



Departamento de Informática

TRABAJO FIN DE GRADO

APLICACIÓN MÓVIL BASADA EN LAS REDES SOCIALES PARA
DAR SOPORTE A LA CORRECCIÓN DE EJERCICIOS PRÁCTICOS
APLICANDO REALIMENTACIÓN MULTI-FUENTE
(PLATAFORMA ANDROID)

Autor: Ángel Ávila Martín-Cocinas

Director: Javier García Guzmán

Tutor: Alberto Heredia García

Leganés, Febrero de 2013

Título: APLICACIÓN MÓVIL BASADA EN LAS REDES SOCIALES PARA DAR SOPORTE A LA CORRECCIÓN DE EJERCICIOS PRÁCTICOS APLICANDO REALIMENTACIÓN MULTI-FUENTE (PLATAFORMA ANDROID).

Autor: Ángel Ávila Martín

Director: Javier García Guzmán

Tutor: Alberto Heredia García

EL TRIBUNAL

Presidente: José María Sierra Cámara

Vocal: Silvia Noemí Santalla Arribas

Secretario: Ángel Lagares Lemos

Realizado el acto de defensa y lectura del Trabajo Fin de Grado el día 10 de Marzo de 2014 en Leganés, en la Escuela Politécnica Superior de la Universidad Carlos III de Madrid, acuerda otorgarle la CALIFICACIÓN de

SECRETARIO

VOCAL

PRESIDENTE

*“La tecnología no es en sí el fin sino el medio
entre la sociedad del conocimiento
y el desarrollo mundial.”*

Agradecimientos

Antes de comenzar, me gustaría dar las gracias a todas aquellas personas que me han ayudado y apoyado durante los últimos años de mi vida y sin los cuales no habría sido posible llegar hasta donde he llegado.

En primer lugar, quiero dar las gracias a mis padres, Ángel y Esperanza, los cuáles siempre me han ayudado y me han dado el apoyo y la educación necesaria para llegar a ser la persona que soy actualmente. Además siempre han permanecido a mi lado, tanto en los buenos como en los malos momentos y sé que siempre podré contar con ellos. Muchísimas gracias.

En segundo lugar, agradecer también a mi hermana, Verónica, con la cuál estoy conviviendo durante los últimos años de carrera y que, al igual que mis padres, siempre me ha dado el apoyo necesario.

Dar las gracias también a todos los compañeros que he conocido durante estos años de universidad. Con ellos he compartido tanto bueno como malos momentos: los ratos en la cafetería, las situaciones de estrés, los exámenes, las prácticas... Espero que tras finalizar la carrera, sigamos en contacto.

Por supuesto, me gustaría dar las gracias a Javier y a Alberto por ofrecerme la oportunidad de realizar este proyecto y por haber estado siempre disponibles para cualquier duda o aclaración que necesitase resolver.

Finalmente, recordar a aquellas personas que tuve a mi lado, pero que por desgracia, ya no están con nosotros.

Resumen

Hoy en día los dispositivos móviles se han convertido en un elemento casi indispensable en nuestras vidas. En pocos años la utilización de este tipo de dispositivos tanto entre los jóvenes como entre los adultos se ha incrementado de una forma que casi ni podíamos llegar a imaginar.

Además, la utilización de las redes sociales ha adquirido también una gran importancia en la vida cotidiana de las personas, ya que son utilizadas tanto para comunicarse con otras personas como para expresar nuestros pensamientos o ideas sobre un tema determinado.

Por otro lado, la evaluación multifuente es aquella en la que la nota obtenida no dependerá únicamente de la valoración dada por el profesor sino que se realizará desde varios puntos de vista, obteniendo de este modo la nota más justa para cada alumno. A pesar del gran número de aplicaciones móviles existentes no encontramos ninguna que permita la evaluación multifuente de alumnos.

Todo ello nos ha llevado al planteamiento de este proyecto, que consistirá en la creación de una aplicación para dispositivos Android que facilitará la corrección de ejercicios prácticos mediante la evaluación multifuente. Gracias a esta aplicación los profesores podrán evaluar a los alumnos, y éstos a su vez podrán evaluar a sus compañeros de grupo. De este modo será posible obtener de forma rápida y sencilla la nota que cada miembro del grupo realmente se merece.

A lo largo de todo este documento se detallará toda la información que ha sido necesaria para la elaboración de este proyecto. Por lo tanto este documento contendrá una pequeña introducción, la especificación de requisitos, el diseño de la aplicación, las pruebas que se deben realizar, la planificación llevada a cabo, el presupuesto y una conclusión final.

Abstract

Nowadays, mobile devices have become in an element almost indispensable in our lives. In a few years, the use of these devices among both young people and adults has increased in a way that we could hardly imagine.

Furthermore, the use of social networks has achieved a great importance in the daily lives of people. These social networks are used to communicate with other people and to express our thoughts and ideas on a given topic.

Otherwise, in multi-source feedback, the grade obtained by the students will not depend only on the teacher's opinion. The grade obtained by the students will depend from several points of view, and each student will obtain the grade he really deserves. Despite the large number of existing mobile applications we haven't found any that allows multi-source feedback.

This has led us to create this project. This project will consist in the creation of an app for Android devices which will facilitate the correction of practical exercises using multi-source feedback. Thanks to this application teachers can assess their students, and students can evaluate their classmates. This application will let us to obtain quickly and easily the grade that each member of the group really deserves.

This document will contain all the necessary for the development of this project. Therefore, this document will contain a little introduction, the requirements specification, design, testing to be performed, conducted planning, budget and a final conclusion.

Índice

Capítulo I. Introducción	20
1.1 Contexto	21
1.2 Problemática y motivación	22
1.3 Objetivos.....	23
1.4 Estructura del documento.....	24
1.5 Glosario de términos.....	26
Capítulo II. Estado del arte	28
2.1 Introducción al tema del proyecto.....	29
2.1.1 Personal Learning Enviroment y Personal Learning Network	30
2.2 Estudio de aplicaciones	32
2.2.1 Cuaderno del profesor.....	33
2.2.2 Profesor ayudante PRO 2.....	33
2.2.3 Evaluación docente alumno/profesor	33
2.2.4 Edmodo.....	34
2.2.5 Grade Book	34
2.2.6 Moodle Mobile.....	34
2.2.7 Comparación de aplicaciones	34
2.3 Entorno de desarrollo móvil	36
2.3.1 Sistema operativo. Android.	38
2.3.2 Lenguaje de programación. Java.	39
2.3.3 Entorno de desarrollo. Eclipse.	40
2.4 Metodología de desarrollo. Scrum.....	40
2.4.1 Ventajas de Scrum.....	41
2.4.2 El proceso.....	41
2.4.3 Roles	42
Capítulo III. Especificación de requisitos.....	44
3.1 Historias de usuario	45
3.1.1 Historias de usuario: Aplicación de profesor.	47
3.1.2 Historias de usuario: Aplicación de alumno.....	51
3.2 Casos de uso	54
3.2.1 Diagrama de casos de uso	56
3.2.2 Casos de uso.....	57
3.3 Requisitos de restricción	63
Capítulo IV. Diseño de la aplicación.....	64
4.1 Alternativas de diseño.....	65
4.1.1 Alternativas de diseño: Número de aplicaciones.....	66

4.1.2 Alternativas de diseño: Almacenamiento de la información.....	67
4.1.3 Alternativas de diseño: Tratamiento de la información	70
4.2 Paper prototype	73
4.2.1 Aplicación de profesor.....	73
4.2.2 Aplicación de alumno	78
4.3 Diagrama de clases.....	82
4.3.1 Diagrama de clases. Aplicación de profesores.....	82
4.3.2 Diagrama de clases. Aplicación de alumnos.	92
4.3.3 Diagrama de clases. Servicio Web.....	100
4.4 Diagramas de secuencia	103
4.4.1 Diagramas de secuencia. Aplicación profesor.	104
4.4.2 Diagramas de secuencia. Aplicación alumno.....	114
4.5 Diseño del servidor FTP	121
Capítulo V. Pruebas de sistema.....	124
5.1 Pruebas de sistema: CU-01.....	126
5.2 Pruebas de sistema: CU-02.....	132
5.3 Pruebas de sistema: CU-03.....	133
5.4 Pruebas de sistema: CU-04.....	136
5.5 Pruebas de sistema: CU-05.....	137
5.6 Pruebas de sistema: CU-06.....	142
5.7 Pruebas de sistema: CU-07.....	145
5.8 Pruebas de sistema: CU-08.....	147
5.9 Pruebas de sistema: CU-09.....	148
5.10 Pruebas de sistema: CU-10.....	149
5.11 Pruebas de sistema: CU-11.....	152
Capítulo VI. Planificación	154
Iteración 1	156
Iteración 2	156
Iteración 3	156
Iteración 4	156
Iteración 5	158
Iteración 6	159
Iteración 7	160
Iteración 8	161
Iteración 9	163
Iteración 10	163
Capítulo VII. Presupuesto.....	164
7.1 Costes de personal	165

7.2 Costes software	165
7.3 Costes hardware	166
7.4 Resumen de costes	166
Capítulo VIII. Conclusión	168
8.1 Resultados obtenidos	169
8.2 Líneas de ampliación	169
8.2.1 Login a través de otras redes sociales	169
8.2.2 Gestionar datos desde la aplicación	170
8.2.3 Notificaciones push	170
8.3 Opinión personal.....	171
Capítulo IX. Bibliografía.....	172
Referencias.....	173

Índice de ilustraciones

Ilustración 1. Evaluación multifuente.....	29
Ilustración 2. Partes de los PLEs	31
Ilustración 3. Venta de dispositivos móviles por sistema operativo.	36
Ilustración 4. Venta de aplicaciones por sistema operativo.	37
Ilustración 5. Android	38
Ilustración 6. Java	39
Ilustración 7. Eclipse	40
Ilustración 8. Metodología scrum.....	42
Ilustración 9. Diagrama de casos de uso	56
Ilustración 10. Alternativa 1. Dropbox.....	68
Ilustración 11. Alternativa 2. Servidor FTP Universidad	69
Ilustración 12. Alternativa 2. Servidor FTP propio.....	69
Ilustración 13. Alternativa 1. Información gestionada por el terminal.....	71
Ilustración 14. Alternativa 2. Información gestionada por un servidor web	72
Ilustración 15. Pantalla de login	74
Ilustración 16. Pantalla de listado de evaluaciones.....	74
Ilustración 17. Pantalla de listado de grupos	75
Ilustración 18. Pantalla de visualización de evaluaciones realizadas	75
Ilustración 19. Pantalla de evaluación	76
Ilustración 20. Pantalla de listado de alumnos.....	76
Ilustración 21. Pantalla de notas de alumno.....	77
Ilustración 22. Pantalla de listado de grupos (Notas)	77
Ilustración 23. Pantalla de notas de grupo	78
Ilustración 24. Pantalla de login	78
Ilustración 25. Pantalla de listado de asignaturas.....	79
Ilustración 26. Pantalla de listado de evaluaciones.....	79
Ilustración 27. Pantalla de visualización de evaluaciones realizadas	80
Ilustración 28. Pantalla de evaluaciones.....	80
Ilustración 29. Pantalla de listado de notas	81
Ilustración 30. Pantalla de notas	81
Ilustración 31. Diagrama de clases. Aplicación profesor	82
Ilustración 32. Diagrama de clases. Aplicación Alumno	92
Ilustración 33. Clase Application.	103
Ilustración 34. DS-CU-01-P	104
Ilustración 35. DS-CU-03-P	105
Ilustración 36. DS-CU-04-P	106
Ilustración 37. DS-CU-05-P	107
Ilustración 38. DS-CU-06-P	108
Ilustración 39. DS-CU-07-P	109
Ilustración 40. DS-CU-08-P	110
Ilustración 41. DS-CU-09-P	111
Ilustración 42. DS-CU-10-P	112
Ilustración 43. DS-CU-11-A.....	113
Ilustración 44. DS-CU-01-A.....	114
Ilustración 45. DS-CU-02-A.....	115
Ilustración 46. DS-CU-03-A.....	116
Ilustración 47. DS-CU-05-A.....	117
Ilustración 48. DS-CU-08-A.....	118

Ilustración 49. DS-CU-10-A.....	119
Ilustración 50. DS-CU-11-A.....	120
Ilustración 51. Estructura FTP.	121

Índice de tablas

Tabla 1. Comparación de aplicaciones.....	35
Tabla 2. Modelo de historias de usuario.....	46
Tabla 3. HU-01-P.....	47
Tabla 4. HU-03-P.....	47
Tabla 5. HU-03-P.....	48
Tabla 6. HU-04-P.....	48
Tabla 7. HU-05-P.....	48
Tabla 8. HU-06-P.....	49
Tabla 9. HU-07-P.....	49
Tabla 10. HU-08-P.....	49
Tabla 11. HU-09-P.....	50
Tabla 12. HU-10-P.....	50
Tabla 13. HU-11-P.....	50
Tabla 14. HU-01-A.....	51
Tabla 15. HU-02-A.....	51
Tabla 16. HU-03-A.....	52
Tabla 17. HU-04-A.....	52
Tabla 18. HU-05-A.....	52
Tabla 19. HU-06-A.....	53
Tabla 20. HU-07-A.....	53
Tabla 21. HU-08-A.....	53
Tabla 22. Modelo de casos de uso.....	54
Tabla 23. CU-01.....	57
Tabla 24. CU-02.....	57
Tabla 25. CU-03.....	58
Tabla 26. CU-04.....	58
Tabla 27. CU-05.....	59
Tabla 28. CU-06.....	60
Tabla 29. CU-07.....	60
Tabla 30. CU-08.....	61
Tabla 31. CU-09.....	61
Tabla 32. CU-10.....	62
Tabla 33. CU-11.....	62
Tabla 34. RR-01.....	63
Tabla 35. RR-02.....	63
Tabla 36. RR-03.....	63
Tabla 37. RR-04.....	63
Tabla 38. Clase Usuario.....	83
Tabla 39. Clase Criterio.....	83
Tabla 40. Clase GrupoCriterios.....	84
Tabla 41. Clase Evaluacion.....	84
Tabla 42. Clase AlumnoNota.....	84
Tabla 43. Clase GrupoCriteriosNota.....	85
Tabla 44. Clase Nota.....	85
Tabla 45. Clase HTTPManager.....	87
Tabla 46. Clase JSONManager.....	88
Tabla 47. Clase DataBaseManager.....	88
Tabla 48. Clase LoginActivity.....	89

Tabla 49. Clase ListadoEvaluacionesActivity	89
Tabla 50. Clase ListadoGruposActivity	90
Tabla 51. Clase EvaluacionActivity	90
Tabla 52. Clase NotasAlumnoActivity	91
Tabla 53. Clase NotasGrupoActivity	91
Tabla 54. Clase Usuario	92
Tabla 55. Clase AlumnoCriterio	93
Tabla 56. Clase Criterio	93
Tabla 57. Clase GrupoCriterios	93
Tabla 58. Clase Evaluacion	94
Tabla 59. Clase AlumnoNota	94
Tabla 60. Clase GrupoCriteriosNota	94
Tabla 61. Clase Nota	95
Tabla 62. Clase HTTPManager	96
Tabla 63. Clase JSONManager	97
Tabla 64. Clase DataBaseManager	97
Tabla 65. Clase LoginActivity	98
Tabla 66. Clase ListadoEvaluacionesActivity	98
Tabla 67. Clase ListadoGruposActivity	99
Tabla 68. Clase EvaluacionActivity	99
Tabla 69. Clase NotasAlumnoActivity	100
Tabla 70. Modelo pruebas	125
Tabla 71. PR-01	126
Tabla 72. PR-02	126
Tabla 73. PR-03	127
Tabla 74. PR-04	127
Tabla 75. PR-05	128
Tabla 76. PR-06	128
Tabla 77. PR-07	129
Tabla 78. PR-08	129
Tabla 79. PR-09	130
Tabla 80. PR-10	130
Tabla 81. PR-11	131
Tabla 82. PR-12	131
Tabla 83. PR-13	132
Tabla 84. PR-14	132
Tabla 85. PR-15	133
Tabla 86. PR-16	133
Tabla 87. PR-17	134
Tabla 88. PR-18	134
Tabla 89. PR-19	135
Tabla 90. PR-20	135
Tabla 91. PR-21	136
Tabla 92. PR-22	136
Tabla 93. PR-23	137
Tabla 94. PR-24	137
Tabla 95. PR-25	138
Tabla 96. PR-26	138
Tabla 97. PR-27	139
Tabla 98. PR-28	139
Tabla 99. PR-29	140
Tabla 100. PR-30	140

Tabla 101. PR-31	141
Tabla 102. PR-32	141
Tabla 103. PR-33	142
Tabla 104. PR-22	142
Tabla 105. PR-35	143
Tabla 106. PR-36	143
Tabla 107. PR-37	143
Tabla 108. PR-35	144
Tabla 109. PR-39	144
Tabla 110. PR-40	145
Tabla 111. PR-41	145
Tabla 112. PR-42	146
Tabla 113. PR-43	146
Tabla 114. PR-44	146
Tabla 115. PR-45	147
Tabla 116. PR-46	147
Tabla 117. PR-47	148
Tabla 118. PR-48	148
Tabla 119. PR-49	149
Tabla 120. PR-50	149
Tabla 121. PR-51	150
Tabla 122. PR-52	150
Tabla 123. PR-53	151
Tabla 124. PR-54	151
Tabla 125. PR-55	152
Tabla 126. PR-56	152
Tabla 127. Planificación. Iteraciones	155
Tabla 128. Costes de personal	165
Tabla 130. Costes software.	165
Tabla 129. Costes hardware	166
Tabla 131. Tabla de costes	166

Capítulo I. Introducción

A lo largo de este documento se realizará una descripción detallada de los pasos seguidos para la realización de la plataforma de aprendizaje que permitirá la corrección de ejercicios y prácticas basadas en la realimentación multifuente. Gracias a esta plataforma se facilitará la corrección de este tipo de ejercicios tanto a los profesores como a los alumnos.

Durante este capítulo en concreto se hablará en primer lugar del contexto existente durante el desarrollo del proyecto y de los problemas existentes en la actualidad y que han llevado a la realización del mismo. A continuación se describirá el objetivo que se pretende alcanzar con el desarrollo de esta aplicación y finalmente se realizará una breve introducción al contenido del resto de secciones que conforman este documento.

1.1 Contexto

En la actualidad, los dispositivos móviles se han convertido en un elemento casi indispensable en la vida de las personas. Se hace difícil imaginar la vida sin este tipo de dispositivos ya que tanto jóvenes como adultos utilizan su teléfono móvil o tablet constantemente ya sea para realizar llamadas, contactar con sus conocidos a través de determinadas aplicaciones, consultar información en internet, jugar, ver películas y realizar casi cualquier cosa que podamos imaginar.

El número de aplicaciones existentes para este tipo dispositivos crece cada día más y pueden encontrarse aplicaciones de todo tipo.

Además de los dispositivos móviles, las redes sociales han ganado también una gran importancia en la vida de las personas. Día a día puede observarse como son cada vez más las personas que hacen uso de ellas para comunicarse con amigos, familiares o conocidos, o incluso para expresar sus pensamientos o sus opiniones acerca de cualquier tema de actualidad.

1.2 Problemática y motivación

En este apartado vamos a hablar de los problemas existen actualmente y de los motivos que nos han llevado a la realización de este proyecto.

En el ámbito universitario la comunicación entre los profesores y los alumnos es de gran importancia ya sea para aclarar dudas en clase, la realización de ejercicios o tareas, las tutorías, la realización de las prácticas y exámenes, y la corrección de los mismos. A pesar de ello, las aplicaciones para este tipo de dispositivos que facilitan esta comunicación son muy limitadas.

Este aspecto es el que ha llevado al planteamiento de este proyecto, que tratará de la evaluación multifuente de los ejercicios prácticos.

La evaluación multifuente es aquella en la que la nota obtenida no dependerá únicamente de la valoración dada por el profesor sino que la evaluación de un alumno se realizará desde al menos tres puntos de vista: el del profesor, el de los compañeros de grupo, y el del propio alumno. Además, una vez completado el proceso de evaluación el evaluado deberá obtener un informe con la evaluación realizada y la nota obtenida.

Por lo tanto, gracias a esta aplicación, los profesores podrán evaluar a los alumnos mediante su tablet y los alumnos podrán evaluarse tanto a sí mismos como a sus compañeros de grupo. Además, los alumnos podrán consultar sus notas de forma inmediata a través de su dispositivo móvil.

De este modo se facilitará en gran medida la realización de este tipo de evaluaciones, ya que todos los cálculos serán realizados de forma automática por la aplicación móvil. Gracias a ello, el profesor se ahorrará una gran cantidad de tiempo, no necesitando pasar sus datos a su ordenador o a una hoja de cálculo, y además se evitarán los errores de cálculo que puedan producirse al calcular la nota de forma manual.

Otro de los aspectos que nos ha llevado a la realización de este proyecto es el ya comentado auge de los dispositivos móviles. Cada día son más los usos que se le dan a este tipo de dispositivo, y por supuesto, no podía faltar entre ellos una aplicación que facilitase la comunicación entre profesores y alumnos.

El hecho de que la aplicación se realice para dispositivos Android supone una motivación extra, ya que actualmente es el líder en ventas del mercado en cuanto a dispositivos móviles se refiere. Además, cada año puede observarse un incremento en la tasa de ventas de estos dispositivos.

1.3 Objetivos

El objetivo principal de este proyecto es la realización de una aplicación que permita la evaluación multifuente de prácticas tanto para profesores como para alumnos. Esta aplicación estará además basada en las redes sociales, utilizando la identidad social del usuario para acceder a las mismas y gestionar los datos correspondientes. La red social elegida para la realización de este proyecto ha sido Twitter debido principalmente a que es la red social más utilizada en la actualidad.

El desarrollo de dicha aplicación constará de tres partes:

- El desarrollo de una aplicación para los alumnos. Dicha aplicación permitirá a los alumnos realizar las siguientes acciones:
 - Realizar la evaluación de las prácticas evaluándose tanto a sí mismo, como a sus compañeros de grupo.
 - Comprobar tanto la nota obtenida por el propio alumno, como la media obtenida por el grupo del que forma parte.
- El desarrollo de una aplicación para los profesores. Dicha aplicación permitirá a los profesores realizar las siguientes acciones:
 - Realizar la evaluación de cada uno de los grupos existentes en cada una de las evaluaciones.
 - Para una evaluación en concreto, comprobar la nota obtenida por cada uno de los grupos.

- Comprobar todas las notas obtenidas por un alumno en concreto.
- El desarrollo de un Servicio Web que permita realizar las siguientes acciones:
 - Gestionar un servidor FTP en el que se encontrarán todos los datos necesarios.
 - Permitir la descarga de los datos necesarios desde el servidor FTP a las aplicaciones de profesores y alumnos.
 - Realizar todas las operaciones y gestiones necesarias para dotar de funcionalidad a las aplicaciones de profesores y alumnos.

1.4 Estructura del documento

Durante este apartado se hablará de las distintas partes de las que constará el presente documento. De este modo, se realizará una pequeña introducción de cada uno de los distintos capítulos.

La estructura de este documento es la que se expone a continuación:

- **Introducción:** Durante este capítulo se hablará en primer lugar del contexto existente durante el desarrollo del proyecto y de los problemas existentes en la actualidad y que han llevado a la realización del mismo. A continuación se describirá el objetivo que se pretende alcanzar con el desarrollo de esta aplicación y finalmente se realizará una breve introducción al contenido del resto de secciones que conforman el documento.
- **Estado del arte:** En este capítulo se realizará en primer lugar una breve introducción al tema del proyecto. Tras ello se realizará un análisis de las aplicaciones móviles y web que existen actualmente y que están relacionadas de algún modo con los objetivos de este proyecto. A continuación se mostrará un análisis del entorno de desarrollo móvil seleccionado. Finalmente se realizará una descripción de la metodología de desarrollo elegida para la realización del proyecto.

- **Especificación de requisitos:** En este capítulo se expondrán de forma detallada los distintos requisitos que debe cumplir la aplicación. Para ello, se utilizará la técnica de historias de usuario y se mostrarán los distintos casos de uso existentes.
- **Diseño de la aplicación:** En este capítulo se analizarán en primer lugar las distintas alternativas de diseño contempladas y se especificará cuál de ellas ha sido seleccionada. A continuación se mostrarán las distintas pantallas de las que constará cada una de las aplicaciones (profesor y alumno). Tras esto se realizará un diagrama de las clases o actividades que forman la aplicación. Finalmente se mostrarán los distintos diagramas de secuencia necesarios para modelar la aplicación.
- **Pruebas de la aplicación:** En este capítulo se detallaran las distintas pruebas realizadas para comprobar el correcto funcionamiento de la aplicación.
- **Planificación:** En este capítulo se mostrará la planificación llevada a cabo durante el proyecto. Puesto que la planificación realizada será una planificación adaptativa, se describirán las distintas iteraciones existentes, los objetivos de cada iteración y las tareas llevadas a cabo para cumplir dichos objetivos.
- **Presupuesto:** A lo largo de este capítulo se realizará una descripción detallada de los costes que ha supuesto la realización de este proyecto. En él se incluirá tanto el coste de personal como el coste del software y el hardware utilizados en el proyecto.
- **Conclusiones:** En este capítulo se mostrarán las conclusiones alcanzadas a partir de la realización del proyecto. En ellas se incluirán los resultados obtenidos en el desarrollo, las futuras líneas de ampliación y mejora de la aplicación y una opinión personal sobre las experiencias y el aprendizaje adquiridos durante su realización.
- **Referencias:** En este apartado se incluirán todas las fuentes utilizadas para la realización de este documento.

1.5 Glosario de términos

Actividad: En Android, una actividad es cada una de las interfaces que muestra una aplicación.

Android: sistema operativo para dispositivos móviles. Está basado en GNU/Linux e inicialmente fue desarrollado por Google.

Emulador: software que permite ejecutar programas o videojuegos en una plataforma diferente de aquella para la cual fueron escritos originalmente. Un emulador trata de modelar de forma precisa el dispositivo de manera que este funcione como si estuviese siendo usado en el aparato original. ^[1]

Evaluación multifuente: proceso que consiste en primer lugar en evaluar a un individuo según sus comportamientos y competencias y desde varias fuentes distintas. A continuación la información de la evaluación es comunicada al evaluado mediante un informe feedback. ^[2]

Dropbox: servicio de alojamiento de archivos multiplataforma en la nube, operado por la compañía Dropbox. El servicio permite a los usuarios almacenar y sincronizar archivos en línea y entre ordenadores y compartir archivos y carpetas con otros.

Hardware: componente físico tecnológico, que trabaja o interactúa de algún modo con la computadora. Incluye elementos internos como el disco duro, CD-ROM, disquetera, e incluso hace referencia a elementos externos como la impresora, el ratón, el teclado, el monitor y otros periféricos.

Metodología ágil: metodología de ingeniería del software basada en el desarrollo iterativo e incremental, donde los requisitos y soluciones evolucionan mediante la colaboración de grupos auto organizados y multidisciplinarios. ^[3]

Nube: Servicio virtual que permite el almacenamiento de archivos a través de internet. Los archivos son alojados en un servidor externo.

Product Backlog: Conjunto de requisitos denominados historias descritos en un lenguaje no técnico y priorizados por valor de negocio, o lo que es lo mismo, por retorno de inversión considerando su beneficio y coste. ^[4]

Servidor FTP: programa que se ejecuta en un servidor conectado normalmente en Internet y que permite el desplazamiento de datos entre diferentes servidores y ordenadores. ^[5]

Servidor Web: programa informático que procesa una aplicación del lado del servidor realizando conexiones con el cliente y generando o cediendo una respuesta para cualquier lenguaje o aplicación del lado del cliente.

Sprint: Iteración de duración prefijada durante la cual el equipo trabaja para convertir las historias del Product Backlog a las que se ha comprometido, en una nueva versión del software totalmente operativo. ^[6]

Software: equipamiento o soporte lógico de un sistema informático que comprende el conjunto de los componentes lógicos necesarios que hacen posible la realización de tareas específicas.

Tablet: computadora portátil con la que se puede interactuar a través de una pantalla táctil. Se trata de un dispositivo a medio camino entre el teléfono móvil y el ordenador portátil.

Capítulo II.

Estado del arte

A lo largo de este capítulo se llevará a cabo el desarrollo del estado del arte.

En primer lugar se realizará una introducción al tema del proyecto, indicando que se quiere desarrollar.

Tras ello se realizará un análisis de las aplicaciones móviles y web que existen actualmente y que están relacionadas de algún modo con los objetivos de este proyecto, analizando las ventajas o inconvenientes de cada una de ellas.

A continuación se mostrará un análisis del entorno de desarrollo móvil seleccionado.

Finalmente se realizará una descripción de la metodología de desarrollo elegida para la realización del proyecto, es decir, una descripción de la metodología de desarrollo ágil Scrum.

2.1 Introducción al tema del proyecto

Este proyecto tiene como objetivo la realización de una aplicación para tablets Android que permita la evaluación multifuente por parte de profesores y alumnos. De este modo, la nota de una práctica no vendrá dada únicamente por la nota otorgada por el profesor, sino que los propios alumnos también podrán evaluarse tanto a sí mismos como a sus compañeros de grupo.

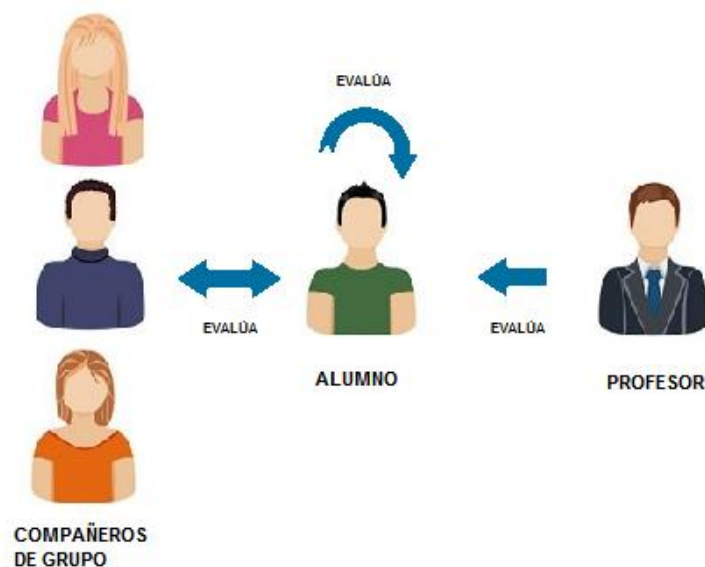


Ilustración 1. Evaluación multifuente.

Gracias a la evaluación multifuente existe más de un punto de vista a la hora de calcular la nota, por lo que será más fácil determinar la nota que un alumno realmente merece. Además, alumnos de un mismo grupo podrán tener diferente nota dependiendo del esfuerzo que haya realizado cada uno de ellos.

Para evitar posibles abusos a la hora de evaluar por parte de los alumnos, la nota calculada por estos deberá estar ponderada con respecto a la del profesor y si es muy superior (o muy inferior) a la calculada por el profesor no se tendrá en cuenta para la media.

Por otro lado, para facilitar el acceso a la aplicación y la comunicación entre sus usuarios se utilizará la red social Twitter. Gracias a esto los usuarios no necesitarán crearse una nueva cuenta en la aplicación sino que en lugar de ello utilizarán un usuario con el que estarán muy familiarizados y desde el que podrán gestionar todas sus asignaturas.

Esta aplicación tendrá un perfil educativo y estará basada en las características de los Personal Learning Enviroment (PLE) y de las Personal Learning Network (PLN).

2.1.1 Personal Leaning Enviroment y Personal Learning Network

Un Entorno Personal de Aprendizaje (PLE) es un enfoque pedagógico potencialmente prometedor tanto para la integración del aprendizaje formal y no formal a través de las redes sociales como para la autorregulación del aprendizaje de los estudiantes. ^{[7] [8]}

Un PLE también puede verse como una colección definida de servicios, herramientas y dispositivos que ayudan a los estudiantes a construir sus redes de conocimientos personales, abarcando tanto nodos de conocimiento tácito (es decir, personas) como nodos de conocimiento explícitos (es decir, información). ^{[7] [9]}

Un PLE debe permitir al alumno: ^{[7] [12]}

- Aprender con otras personas, gestionando las relaciones con sus profesores y compañeros y creando nuevas relaciones con personas que no pertenecen a su red de aprendizaje habitual.
- Controlar sus recursos de aprendizaje, lo que permite estructurar y compartir los recursos que han conseguido.
- Administrar las actividades en las que participan, teniendo la oportunidad de crear o unirse a actividades como, por ejemplo, grupos de estudios.
- Integrar su aprendizaje, dándoles la oportunidad de combinar los aprendizajes obtenidos y de establecer vínculos entre el aprendizaje formal y el no formal.

Estos entornos de aprendizaje presentan una estructura similar a la que presenta una asignatura a lo largo de todo el curso. De este modo, estos sistemas están formados por distintos componentes que permitirán la preparación de la asignatura, la impartición de clases, el seguimiento de una clase, el estudio de la asignatura o la evaluación de la asignatura. ^[13]

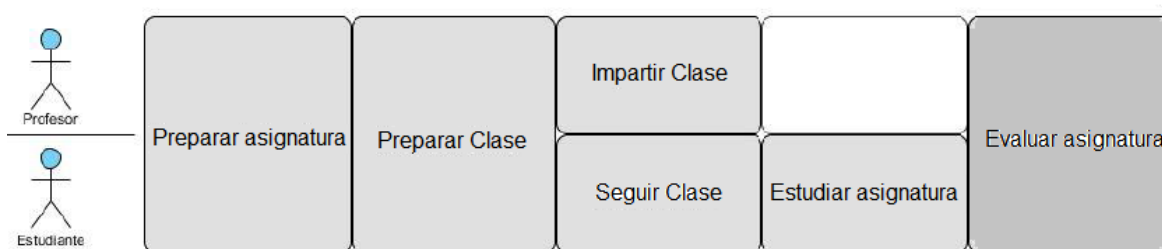


Ilustración 2. Partes de los PLEs ^[13]

A lo largo de este proyecto nos centraremos exclusivamente en la parte de la evaluación de la asignatura, que permitirá a profesores y a alumnos evaluar una asignatura y comprobar la nota obtenida tras la evaluación.

Paralelamente al concepto de los Entornos Personales de Aprendizaje encontramos otro aparentemente muy similar: Red Personal de Aprendizaje. Una Red Personal de Aprendizaje (o PLN) consiste en el conjunto de herramientas que nos permiten buscar o publicar determinados contenidos, los recursos utilizados para ello (aplicaciones web, blogs, páginas web...), las fuentes de información y las aplicaciones o webs que permitan el contacto entre personas (Twitter, Facebook...) y que son utilizadas para alcanzar las metas vinculadas con la adquisición de nuevas competencias. ^{[10][11]}

2.2 Estudio de aplicaciones

En este capítulo se realizará una búsqueda de aplicaciones que tengan ciertas similitudes con la aplicación que se quiere realizar en este proyecto. Una vez encontradas se realizará un análisis de las mismas, mostrando su descripción, describiendo que nos permiten hacer, comparándolas con la aplicación a realizar y mostrando sus ventajas y sus inconvenientes.

La búsqueda de estas aplicaciones se realizará en primer lugar a través de Google Play y a continuación se realizará una búsqueda de aplicaciones a través de la web.

Durante la búsqueda en Play Store se realizará una búsqueda a través de la categoría “Educación” y a través de los criterios de búsqueda “profesor”, “evaluación”, “docencia” y “educación”.

Las aplicaciones encontradas en Play Store a través de los criterios de búsqueda anteriormente citados han sido las siguientes:

- Cuaderno del profesor.
- Profesor ayudante PRO 2
- Evaluación docente alumno y Evaluación docente profesor

Para la búsqueda a través de la web se utilizarán como criterios de búsqueda “aplicaciones android docentes”, “aplicaciones android para profesores” y “aplicaciones android evaluación”.

Mediante estos criterios de búsqueda las aplicaciones encontradas son las siguientes:

- Edmodo
- Grade Book
- Moodle Mobile.

A continuación se profundizará en cada una de las aplicaciones mencionadas.

2.2.1 Cuaderno del profesor

Cuaderno del Profesor es una aplicación que tiene como objetivo agilizar y optimizar las tareas docentes habituales.

La aplicación está diseñada para sustituir los cuadernos tradicionales del profesorado en formato papel, que se emplean hasta ahora para gestionar los datos de cursos y alumnado. Es por tanto una herramienta para ser usada durante el transcurso de la clase, ahorrando mucho tiempo fuera ella.

Esta aplicación permite al profesor importar datos de alumnos desde un PC, calificar a los alumnos, llevar el control de asistencia, almacenar fichas con información de cada alumno, configurar el horario de clases, exportar los datos en formato CVS y crear copias de seguridad de los datos.

2.2.2 Profesor ayudante PRO 2

Profesor ayudante PRO 2 es una aplicación diseñada para Android 4.0 o superior y está dedicada completamente a los profesores.

Esta aplicación permite llevar un seguimiento de la asistencia de los alumnos, enviar correos electrónicos a los distintos grupos de alumnos, notificar a los padres del estudiante cuando estos se encuentran en clase, y llevar a cabo una planificación de las clases.

2.2.3 Evaluación docente alumno/profesor

Estas aplicaciones han sido desarrolladas por la Universidad Tecnológica Metropolitana (UTM). Por un lado, la aplicación evaluación docente alumnos permite a los alumnos evaluar a sus profesores. Por el otro, la aplicación evaluación docente profesor permite a los profesores realizar la evaluación de todos sus alumnos.

2.2.4 Edmodo

Edmodo es una plataforma libre de aprendizaje social-docente centrada en la educación. Ofrece libre acceso a herramientas para la gestión del aula, la comunicación y el aprendizaje en línea.

Esta herramienta proporciona una manera segura y sencilla a los profesores y los estudiantes para participar en las distintas discusiones, debates o dudas que puedan surgir en torno a una clase. Además permite la visualización de las asignaciones, el seguimiento de las calificaciones y el intercambio de archivos.

2.2.5 Grade Book

Gradebook es una aplicación diseñada para los alumnos que permite llevar un seguimiento de su curso académico. Gracias a esta aplicación los alumnos podrán llevar un seguimiento de las calificaciones obtenidas en cada asignatura y de la media ponderada obtenida.

2.2.6 Moodle Mobile

Moodle Mobile es la aplicación oficial de Moodle y está diseñada tanto para móviles como para tablets. Es una aplicación basada en HTML5 que utiliza las tecnologías web más comunes y permite conectar con los distintos servidores Moodle. Básicamente, es un cliente de servicios web que utiliza REST como protocolo para obtener y enviar información.

Esta aplicación permite subir archivos privados, enviar mensajes a los participantes de un curso, acceder a los contenidos del curso, descargar recursos, evaluar a los alumnos y ver información de otros alumnos.

2.2.7 Comparación de aplicaciones

A continuación se mostrará una comparación de las aplicaciones en función de los puntos que se muestran a continuación:

- Permiten la evaluación de alumnos.
- Permiten la evaluación multifuente.
- Permiten visualizar la nota de los alumnos a los profesores.
- Permiten visualizar la nota de los alumnos a los alumnos.
- Utilizan las redes sociales.

En la siguiente tabla se incluyen todos estos puntos:

Aplicación	Evaluación alumnos	Evaluación multifuente	Visualizar notas prof.	Visualizar notas alum.	Redes sociales
Cuaderno del profesor	SI	NO	SI	NO	NO
Profesor ayudante PRO	SI	NO	SI	NO	NO
Evaluación docente	SI	NO	SI	NO	NO
Edmodo	SI	NO	SI	SI	NO
GradeBook	NO	NO	NO	SI	NO
Moodle Mobile	SI	NO	SI	SI	NO

Tabla 1. Comparación de aplicaciones

Como podemos observar, ninguna de las aplicaciones encontradas permite la evaluación multifuente, que es el objetivo principal de este proyecto.

Además, ninguna de ellas hace uso de las redes sociales, y muchas de ellas ni siquiera pueden ser utilizadas por los alumnos.

Por tanto podemos concluir este apartado diciendo que ninguna de las aplicaciones analizadas cumple con los requisitos establecidos para la realización de este proyecto.

2.3 Entorno de desarrollo móvil

Actualmente los dispositivos móviles se han convertido en elementos indispensables de la vida cotidiana y la venta y utilización de estos dispositivos se ha incrementado en gran medida durante los últimos años.

Existen dos sistemas operativos que están compitiendo por ser los líderes del mercado en la venta de este tipo de dispositivos. Estos sistemas operativos son Android y IOS. A nivel global, los fabricantes que apuestan por Android se llevan un 70% de las ventas, y el iPhone se queda en un 21% del mercado. Una clara supremacía del sistema android. Además hay muchos fabricantes desarrollando smartphones Android, mientras que iOS está disponible únicamente para iPhone.

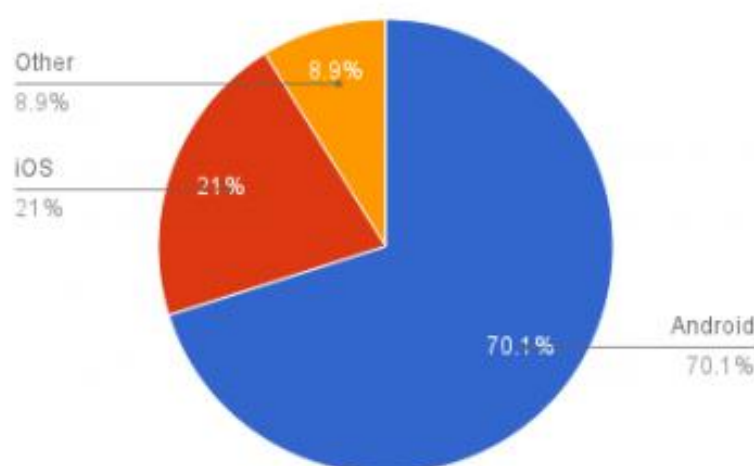


Ilustración 3. Venta de dispositivos móviles por sistema operativo. ^[14]

En cuanto al número de aplicaciones descargadas, los dos mayores competidores vuelven a ser Android y IOS. Para el primer trimestre de 2013, prácticamente la mitad de todas las descargas (un 51%) corresponden a Android, mientras que iOS se quedaría con un 40%, y el resto de sistemas se llevan el 9% restante.

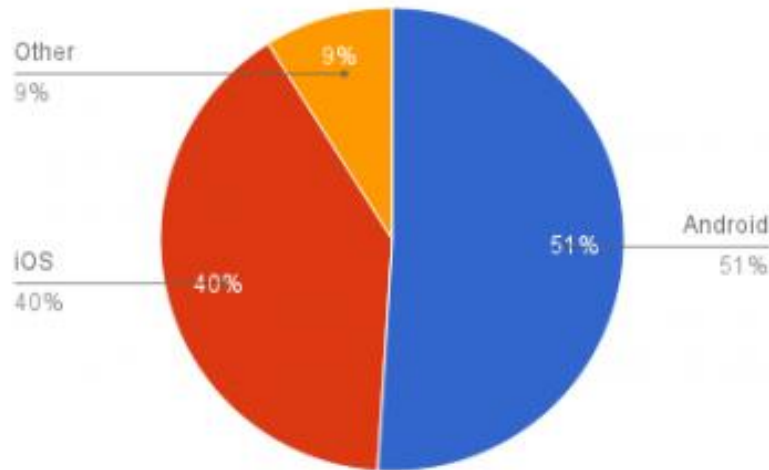


Ilustración 4. Venta de aplicaciones por sistema operativo. ^[14]

Como podemos observar en los gráficos mostrados anteriormente, el líder tanto en la venta de dispositivos móviles como en la descarga de aplicaciones móviles es Android.

Android está siendo el sistema operativo que mayor crecimiento está experimentando en los últimos años y además se espera que en los próximos su crecimiento se mantenga en los mismos niveles que actualmente, con lo que la diferencia entre ambos competidores podría llegar a ser incluso mayor.

Este, entre muchos otros, ha sido uno de los principales motivos que nos han llevado a que la aplicación realizada sea para este sistema operativo.

2.3.1 Sistema operativo. Android.



Ilustración 5. Android

Android es un sistema operativo basado en Linux diseñado principalmente para dispositivos móviles con pantalla táctil, como teléfonos inteligentes o tabletas, aunque también puede ser utilizado en ordenadores portátiles, netbooks, relojes de pulsera, auriculares y otros dispositivos. ^[15]

Android es considerado como uno de los modelos de negocio más exitosos, pues su desarrollo estratégico contempla los factores que más se tienen en cuenta dentro de las herramientas y metodologías desarrollados por expertos en negocios. Este sistema operativo se ha convertido en un modelo a seguir por desarrolladores de tendencias y negocios de alto impacto.

Android, al contrario que otros sistemas operativos para dispositivos móviles como iOS o Windows Phone, se desarrolla de forma abierta y se puede acceder tanto al código fuente como a la lista de incidencias donde se pueden ver problemas aún no resueltos y reportar problemas nuevos.

Las aplicaciones se desarrollan habitualmente en el lenguaje Java junto con el Android Software Development Kit (Android SDK). Sin embargo, también están disponibles otras herramientas de desarrollo, incluyendo un Kit de Desarrollo Nativo para aplicaciones o extensiones en C o C++.

El desarrollo de aplicaciones para Android no requiere aprender lenguajes complejos de programación. Todo lo que se necesita es un conocimiento aceptable de Java y estar en posesión del kit de desarrollo de software o «SDK» provisto por Google, el cual se puede descargar gratuitamente.

Todas las aplicaciones están comprimidas en formato APK, que se pueden instalar sin dificultad desde cualquier explorador de archivos en la mayoría de dispositivos. La mayor parte de ellas están disponibles mediante Google Play, que es la tienda en línea

de software desarrollado por Google para dispositivos Android. La aplicación llamada "Play Store", que se encuentra instalada en la mayoría de los dispositivos Android, permite a los usuarios navegar y descargar aplicaciones publicadas por los desarrolladores.

2.3.2 Lenguaje de programación. Java.



Ilustración 6. Java

Aunque como ya hemos dicho anteriormente existen varios lenguajes que nos permiten la programación de aplicaciones Android, el más extendido de todos ellos es Java, por lo que será el elegido para la programación de la aplicación llevada a cabo en este proyecto.

El lenguaje de programación Java fue originalmente desarrollado por James Gosling y publicado en 1995 como un componente fundamental de la plataforma Java de Sun Microsystems. Su sintaxis deriva mucho de C y C++, pero tiene menos facilidades de bajo nivel que cualquiera de ellos. Las aplicaciones de Java son generalmente compiladas a bytecode (clase Java) que puede ejecutarse en cualquier máquina virtual Java (JVM) sin importar la arquitectura de la computadora subyacente. ^[16]

Es un lenguaje de programación de propósito general, concurrente, orientado a objetos y basado en clases que fue diseñado específicamente para tener tan pocas dependencias de implementación como fuera posible. Su intención es permitir que los desarrolladores de aplicaciones escriban el programa una vez y lo ejecuten en cualquier dispositivo, lo que quiere decir que el código que es ejecutado en una plataforma no tiene que ser recompilado para correr en otra. Java es uno de los lenguajes de programación más populares en uso, particularmente para aplicaciones de cliente-servidor de web. ^[16]

2.3.3 Entorno de desarrollo. Eclipse.



Ilustración 7. Eclipse

La gran mayoría de aplicaciones de Android deben su existencia a Eclipse. Eclipse es un IDE de código abierto (entorno de desarrollo integrado) para los proyectos de Java. Básicamente, es el lugar donde se realiza el software de la aplicación, siendo apoyada por diversas etapas de su ciclo de vida. Google apoya oficialmente este entorno, y ha creado el Android Development Tools plugin para Eclipse. ^[17]

Gracias a este plugin Eclipse permite crear código orientado a Android. Además cuenta también con el apoyo de emuladores que muestran cómo el código se ejecutaría en versiones estandarizadas de determinados dispositivos Android.

2.4 Metodología de desarrollo. Scrum.

Para la realización de este proyecto llevaremos a cabo una planificación basada en Scrum.

Scrum es un marco de trabajo para la gestión y desarrollo de software basada en un proceso iterativo e incremental utilizado comúnmente en entornos basados en el desarrollo ágil de software. Emplea una aproximación iterativa e incremental para optimizar la predictibilidad y controlar el riesgo y está basado en la transparencia, la inspección y la adaptación. ^[18]

Esta metodología de trabajo promueve la innovación, motivación y compromiso del equipo que forma parte del proyecto, por lo que los profesionales encuentran un ámbito propicio para desarrollar sus capacidades.

2.4.1 Ventajas de Scrum

Las principales ventajas que presenta esta metodología son las siguientes: ^[19]

- **Flexibilidad a cambios:** Alta capacidad de reacción ante los cambios de requerimientos generados por necesidades del cliente o evoluciones del mercado. La metodología está diseñada para adaptarse a los cambios de requerimientos que conllevan los proyectos complejos.
- **Mayor calidad del software:** La metódica de trabajo y la necesidad de obtener una versión funcional después de cada iteración, ayuda a la obtención de un software de calidad superior.
- **Mayor productividad:** Se consigue, entre otras razones, gracias a la eliminación de la burocracia y a la motivación del equipo que proporciona el hecho de que sean autónomos para organizarse.
- **Predicciones de tiempos:** Mediante esta metodología se conoce la velocidad media del equipo por sprint, con lo que consecuentemente, es posible estimar fácilmente para cuando se dispondrá de una determinada funcionalidad.
- **Reducción de riesgos:** El hecho de llevar a cabo las funcionalidades de más valor en primer lugar y de conocer la velocidad con la que el equipo avanza en el proyecto, permite despejar riesgos eficazmente de manera anticipada.

2.4.2 El proceso

^[19] En Scrum el desarrollo se realiza de forma iterativa e incremental. Cada iteración, denominada sprint, tiene una duración preestablecida de entre 2 y 4 semanas, obteniendo como resultado una versión del software con nuevas prestaciones listas para ser usadas. En cada nuevo Sprint, se va ajustando la funcionalidad ya construida y se añaden nuevas prestaciones priorizándose siempre aquellas que aporten mayor valor de negocio.

El conjunto de características que forma parte de cada sprint viene dado en el Product Backlog, que es un conjunto de requisitos de alto nivel priorizados que definen

el trabajo a realizar. Los elementos del Product Backlog que forman parte del sprint se determinan durante la reunión de Sprint Planning. Durante el sprint, nadie puede cambiar el Sprint Backlog, lo que significa que los requisitos están congelados durante el sprint.

Al final de cada sprint, el equipo deberá presentar los avances logrados, y el resultado obtenido es un producto potencialmente entregable al cliente. Asimismo, se recomienda no agregar objetivos al sprint o al sprint backlog a menos que la falta de estos objetivos amenace al éxito del proyecto. La constancia permite la concentración y mejora la productividad del equipo de trabajo.

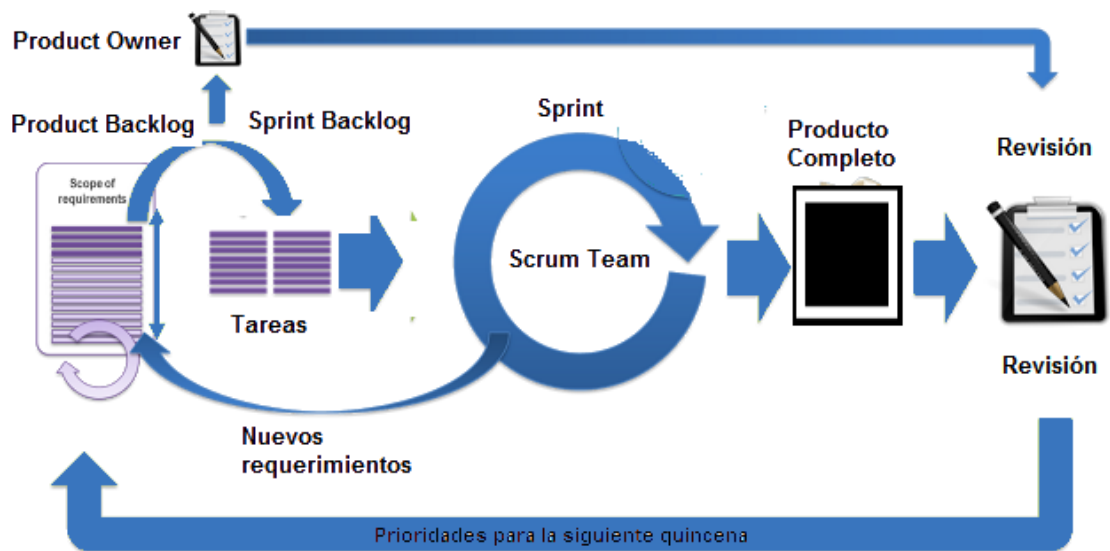


Ilustración 8. Metodología scrum

2.4.3 Roles

En esta metodología de desarrollo existen únicamente 3 roles: ^[18]

- **Product Owner:** Es el representante de todos los interesados del producto final. Sus funciones son marcar las prioridades del producto, llevar el control de las estimaciones y asegurarse de que el equipo Scrum trabaje de forma adecuada desde la perspectiva del negocio. Además es el encargado de escribir las historias de usuario y priorizarlas.

- **Scrum Master:** Es el responsable del proceso de Scrum y sus funciones son la incorporación de Scrum en la cultura de la organización, asegurar el cumplimiento de los roles y responsabilidades y la formación y entrenamiento en el proceso. Su principal objetivo es eliminar los obstáculos que impiden que el equipo alcance el objetivo del sprint. El ScrumMaster no es el líder del equipo (porque ellos se auto-organizan), sino que actúa como una protección entre el equipo y cualquier influencia que le distraiga.
- **Scrum Team:** Grupo de profesionales con los conocimientos técnicos necesarios, que desarrollan el proyecto de manera conjunta llevando a cabo las historias a las que se comprometen al inicio de cada sprint.

Capítulo III.

Especificación de requisitos

En este tercer capítulo se mostrarán los requisitos que debe cumplir la aplicación.

Para ello se hará uso en primer lugar de las historias de usuario, que indicarán los requisitos y funcionalidades que debe cumplir la aplicación a partir de lo expresado por el cliente.

Tras esto, se mostrarán los distintos casos de uso establecidos a partir de las historias de usuario, determinando cuales son los pasos a seguir para llevar a cabo cada uno de ellos.

Finalmente, se especificarán cuáles son los requisitos de restricción que tendrá la aplicación, es decir, las restricciones que se deben tener en cuenta a la hora de implementar o utilizar la aplicación.

3.1 Historias de usuario

Las historias de usuario son la representación de los distintos requisitos software escritos en un lenguaje sencillo y entendible por el usuario. Son una forma de describir una tarea concisa que aporta valor al usuario o al negocio. ^[20]

Las historias de usuario definen lo que se debe construir en el proyecto software, y deben ser independientes unas de otras, negociables, estimables, pequeñas y verificables.

Este tipo de captura de requisitos es utilizado en las metodologías de desarrollo ágiles debido principalmente a que permite dividir el proyecto en pequeñas entregas y a que permite estimar fácilmente el esfuerzo de desarrollo.

Como normal general, las historias de usuario deben responder a las siguientes preguntas: ¿Quién se beneficia?, ¿qué se quiere? y ¿cuál es el beneficio?

El modelo que utilizaremos para representar las distintas historias de usuario será el siguiente:

Identificador	
Título	
Prioridad	
Aplicación	
Descripción	
Validación	

Tabla 2. Modelo de historias de usuario.

A continuación se detalla el contenido de cada uno de estos campos:

Identificador: Etiqueta que representa una historia de usuario de forma única e inequívoca. El identificador presentará el formato HU-XX-Y, donde XX representan los números comenzando desde el 01 y continuando hasta que sea necesario, e Y representa la aplicación a la que hace referencia, pudiendo tomar el valor 'P' (para la aplicación del profesor) o el valor 'A' (para la aplicación del alumno).

Título: Nombre breve y descriptivo para cada historia de usuario.

Prioridad: Determina la prioridad que tiene cada una de las historias de usuario. Esta prioridad podrá ser "Alta", "Media" o "Baja".

Aplicación: Este valor hace referencia a la aplicación a la que se refiere la historia de usuario. Por tanto, podrá tomar los valores "Profesor" o "Alumno".

Descripción: En este campo se incluirá la descripción de la funcionalidad dada por el usuario. En él se especificará con un lenguaje común para el usuario que es lo que debe realizar la aplicación.

Validación: Resultado que esperamos obtener una vez seguidos los pasos indicados anteriormente y que servirá para determinar cuando la historia de usuario está correctamente implementada.

A continuación se mostrarán las historias de usuario obtenidas para cada una de las aplicaciones (Profesor y alumno).

3.1.1 Historias de usuario: Aplicación de profesor.

En este apartado se mostrarán las distintas historias de usuario que deben llevarse a cabo en la aplicación de los profesores.

Identificador	HU-01-P
Título	Loguearse en la aplicación.
Prioridad	Alta
Aplicación	Profesor
Descripción	El profesor deberá poder acceder a la aplicación mediante una cuenta de Twitter. La aplicación deberá comprobar si la cuenta utilizada se corresponde con una cuenta de profesor.
Validación	La aplicación deberá validar que el usuario y la contraseña introducida son correctos. Además, mostrará un aviso cuando la cuenta introducida no pertenezca a una cuenta de profesor.

Tabla 3. HU-01-P

Identificador	HU-02-P
Título	Listar las evaluaciones disponibles.
Prioridad	Alta
Aplicación	Profesor
Descripción	El profesor deberá ser capaz de ver un listado con todas las evaluaciones de su asignatura.
Validación	Una vez introducidos el usuario y la contraseña de forma correcta, la aplicación deberá mostrar un listado con todas las evaluaciones disponibles para la asignatura a la que se refiere dicha cuenta en Twitter.

Tabla 4. HU-03-P

Identificador	HU-03-P
Título	Mostrar el estado de la evaluación.
Prioridad	Baja
Aplicación	Profesor
Descripción	El profesor deberá visualizar si la evaluación está abierta o cerrada.
Validación	Tras loguearse en la aplicación y visualizarse el listado de las evaluaciones, se visualizará para todas ellas si dichas evaluaciones se encuentran abiertas o cerradas.

Tabla 5. HU-03-P

Identificador	HU-04-P
Título	Cerrar evaluación.
Prioridad	Baja
Aplicación	Profesor
Descripción	El profesor deberá ser capaz de cerrar una evaluación para que los alumnos no puedan evaluarse a partir de ese momento.
Validación	Una vez cerrada la aplicación, se comprobará que la evaluación en el listado de evaluaciones aparece como “Cerrada”.

Tabla 6. HU-04-P

Identificador	HU-05-P
Título	Mostrar listado de grupos.
Prioridad	Alta
Aplicación	Profesor
Descripción	El profesor deberá ver que grupos forman parte de cada evaluación.
Validación	Una vez seleccionada una evaluación se comprobará que se muestran todos los grupos pertenecientes a dicha evaluación y a dicha asignatura.

Tabla 7. HU-05-P

Identificador	HU-06-P
Título	Grupos evaluados.
Prioridad	Media
Aplicación	Profesor
Descripción	El profesor deberá ser capaz de ver que grupos han sido evaluados y cuáles no.
Validación	Al aparecer el listado de grupos, la aplicación deberá diferenciar entre aquellos grupos para los que el profesor ya ha realizado la evaluación y aquellos para los que aún no ha sido realizada.

Tabla 8. HU-06-P

Identificador	HU-07-P
Título	Realizar evaluación.
Prioridad	Alta
Aplicación	Profesor
Descripción	El profesor deberá ser capaz de evaluar a todos los grupos existentes en cada una de las evaluaciones.
Validación	Una vez seleccionado el grupo que se desea evaluar, aparecerá un listado de criterios que podrán seleccionarse o deseleccionarse en función de si el grupo ha cumplido o no con el criterio.

Tabla 9. HU-07-P

Identificador	HU-08-P
Título	Ver evaluación realizada.
Prioridad	Alta
Aplicación	Profesor
Descripción	El profesor deberá ser capaz de ver las evaluaciones que ha realizado y que valores ha dado en cada una de ellas.
Validación	Una vez seleccionada una evaluación ya realizada el profesor verá los datos que ha introducido anteriormente.

Tabla 10. HU-08-P

Identificador	HU-09-P
Título	Ver notas alumnos.
Prioridad	Alta
Aplicación	Profesor
Descripción	El profesor deberá ser capaz de ver un listado de todos los alumnos y para cada uno de ellos podrá ver todas las notas que ha obtenido.
Validación	Al seleccionar un alumno determinado, el profesor visualizará en la aplicación un listado de todas las evaluaciones para las que el alumno tiene una nota disponible y la nota obtenida en cada una de ellas.

Tabla 11. HU-09-P

Identificador	HU-10-P
Título	Ver notas evaluaciones.
Prioridad	Alta
Aplicación	Profesor
Descripción	El profesor deberá ser capaz de ver un listado con la nota de todos los grupos para una determinada evaluación.
Validación	Al seleccionar una evaluación se visualizarán todas las notas de todos los grupos para la evaluación seleccionada.

Tabla 12. HU-10-P

Identificador	HU-11-P
Título	Cerrar sesión.
Prioridad	Media
Aplicación	Profesor
Descripción	El profesor deberá poder cerrar sesión para loguearse con otra cuenta.
Validación	La aplicación deberá permitir cerrar la sesión. Una vez cerrada la aplicación mostrará nuevamente la pantalla de login.

Tabla 13. HU-11-P

3.1.2 Historias de usuario: Aplicación de alumno.

En este apartado se mostrarán las distintas historias de usuario que deben llevarse a cabo en la aplicación de los alumnos.

Identificador	HU-01-A
Título	Loguearse en la aplicación.
Prioridad	Media
Aplicación	Alumno
Descripción	El alumno deberá poder acceder a la aplicación mediante una cuenta de Twitter. La aplicación deberá comprobar si la cuenta utilizada se corresponde con una cuenta de alumno.
Validación	La aplicación deberá validar que el usuario y la contraseña introducida son correctos. Además, mostrará un aviso cuando la cuenta introducida no pertenezca a una cuenta de alumno.

Tabla 14. HU-01-A

Identificador	HU-02-A
Título	Listar las asignaturas disponibles.
Prioridad	Alta
Aplicación	Alumno.
Descripción	El alumno deberá ser capaz de ver un listado con todas sus asignaturas.
Validación	Una vez introducidos el usuario y la contraseña de forma correcta, la aplicación deberá mostrar un listado con todas las asignaturas disponibles para el alumno.

Tabla 15. HU-02-A

Identificador	HU-03-A
Título	Listar las evaluaciones disponibles.
Prioridad	Alta
Aplicación	Alumno
Descripción	El alumno deberá ser capaz de ver un listado con todas las evaluaciones disponibles para la asignatura seleccionada.
Validación	Una vez seleccionada una asignatura la aplicación deberá mostrar un listado con todas las evaluaciones de dicha asignatura.

Tabla 16. HU-03-A

Identificador	HU-04-A
Título	Mostrar el estado de la evaluación.
Prioridad	Baja
Aplicación	Alumno
Descripción	El alumno deberá ver si la evaluación está abierta, cerrada, evaluada o no evaluada.
Validación	Tras seleccionar una asignatura y ver el listado de evaluaciones, la aplicación deberá diferenciar entre aquellas evaluaciones que se encuentren abiertas y aquellas que se encuentren cerradas, y entre aquellas que ya hayan sido realizadas y aquellas que no.

Tabla 17. HU-04-A

Identificador	HU-05-A
Título	Realizar evaluación.
Prioridad	Alta
Aplicación	Alumno
Descripción	El alumno deberá ser capaz de evaluar tanto a sus compañeros de grupo como a él mismo para cada una de las evaluaciones.
Validación	Una vez seleccionada una evaluación el alumno verá un listado de criterios y otro de compañeros de grupo a lo que podrá evaluar.

Tabla 18. HU-05-A

Identificador	HU-06-A
Título	Ver evaluación realizada.
Prioridad	Alta
Aplicación	Alumno
Descripción	El alumno deberá ser capaz de ver las evaluaciones que ha realizado y que valores ha dado en cada una de ellas.
Validación	Una vez seleccionada una evaluación ya realizada el alumno verá los datos que ha introducido anteriormente.

Tabla 19. HU-06-A

Identificador	HU-07-A
Título	Ver notas.
Prioridad	Alta
Aplicación	Alumno
Descripción	El alumno deberá ser capaz de ver un listado de todas las notas que ha obtenido.
Validación	Para cada evaluación, la aplicación mostrará la calificación obtenida por el alumno.

Tabla 20. HU-07-A

Identificador	HU-08-A
Título	Cerrar sesión.
Prioridad	Media
Aplicación	Alumno
Descripción	El alumno deberá poder cerrar sesión para loguearse con otra cuenta.
Validación	La aplicación deberá permitir cerrar la sesión. Una vez cerrada, la aplicación mostrará nuevamente la pantalla de login.

Tabla 21. HU-08-A

3.2 Casos de uso

En este apartado se describirán los casos de uso obtenidos a partir de las historias de usuario del apartado anterior.

Un caso de uso es una descripción de los pasos o las actividades que deberán realizarse para llevar a cabo algún proceso. Los personajes o entidades que participarán en un caso de uso se denominan actores. En el contexto de ingeniería del software, un caso de uso es una secuencia de interacciones que se desarrollarán entre un sistema y sus actores en respuesta a un evento que inicia un actor principal sobre el propio sistema. ^[21]

Los diagramas de casos de uso sirven para especificar la comunicación y el comportamiento de un sistema mediante su interacción con los usuarios y otros sistemas, o lo que es lo mismo, un diagrama que muestra la relación entre los actores y los casos de uso en un sistema. Los diagramas de casos de uso se utilizan para ilustrar los requerimientos del sistema al mostrar cómo reacciona a eventos que se producen en su ámbito o en él mismo. ^[21]

El modelo que se utilizará para la representación de cada uno de los casos de uso será el siguiente:

Identificador	
Título	
Actores	
Descripción	
Precondiciones	
Postcondiciones	
Flujo principal	
Historia de usuario	

Tabla 22. Modelo de casos de uso

A continuación se detalla el contenido de cada uno de estos campos:

Identificador: Etiqueta que representa un caso de uso de forma única e inequívoca. El identificador presentará el formato CU-XX, donde XX representa los números naturales comenzando desde el 01 y continuando hasta que sea necesario para completar todos los casos de uso existentes.

Título: Nombre breve y descriptivo para cada caso de uso que dará una breve información sobre dicho caso de uso.

Actores: Son personas, organizaciones o sistemas externos que desempeñan un papel en una o más interacciones con el sistema con el fin de lograr un objetivo. También puede considerarse actores todo aquello que inicia un caso de uso. En nuestro caso, el actor de un caso de uso podrá ser un “Profesor” o un “Alumno”.

Descripción: Este campo incluirá la descripción del caso de uso. En él se especificará qué debe suceder cuando el actor interactúa con el sistema.

Precondiciones: Condiciones que deben darse previamente para que el caso de uso pueda ser llevado a cabo.

Postcondiciones: Condiciones que deben cumplirse una vez se ha completado el caso de uso.

Flujo principal: Conjunto de pasos que ha de seguir el usuario para completar el caso de uso.

Historia de usuario: Historias de usuario con las que está relacionada el caso de uso y que han llevado a su creación.

3.2.1 Diagrama de casos de uso

A continuación se mostrará el diagrama de casos de uso de nuestro sistema. Gracias al diagrama de casos de uso podemos ver de forma clara los distintos casos de uso existentes en la aplicación y los actores que intervienen en ellos.

El diagrama de casos de uso de nuestro sistema es el que sigue a continuación:

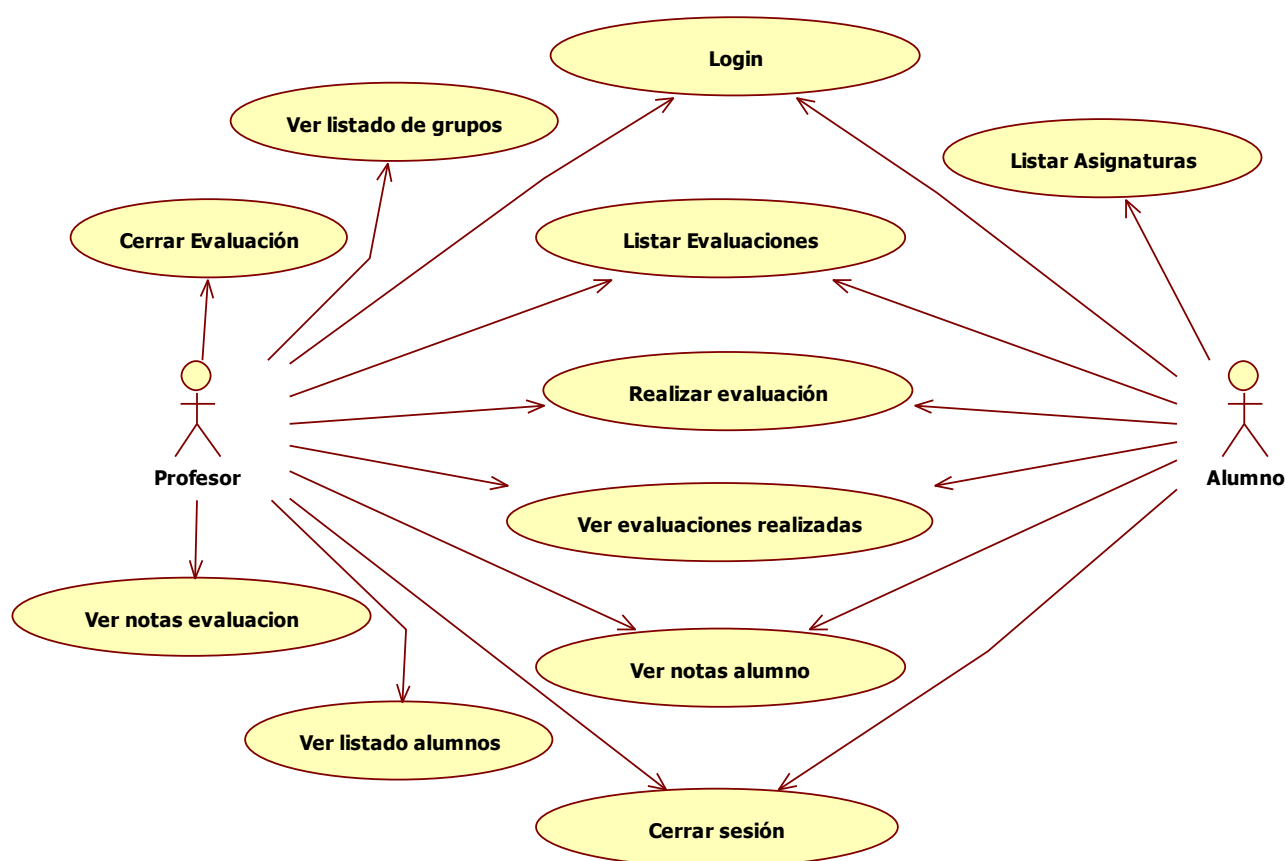


Ilustración 9. Diagrama de casos de uso

Como podemos observar, los casos de uso existentes son muy variados, y no son los mismos para profesores y para alumnos. Además debe tenerse en cuenta que un mismo caso de uso, puede hacer referencia a más de una historia de usuario.

Los casos de uso existentes quedarán detallados en el siguiente apartado.

3.2.2 Casos de uso

En este punto se mostrarán los distintos casos de uso existentes. Para ello se hará uso la tabla modelo especificada anteriormente.

Identificador	CU-01
Título	Login
Actores	Profesor y Alumno
Descripción	Los usuarios deberán poder entrar en la aplicación haciendo uso de su nombre de usuario en la red social Twitter.
Precondiciones	-
Postcondiciones	La aplicación comprueba el usuario y la contraseña y el usuario accede a la aplicación correctamente.
Flujo principal	<ol style="list-style-type: none">1. Abrir la aplicación.2. Introducir el usuario y su contraseña de Twitter.3. Pulsar el botón aceptar.
Historia de usuario	HU-01-P, HU-01-A.

Tabla 23. CU-01

Identificador	CU-02
Título	Listar asignaturas
Actores	Alumno
Descripción	Los alumnos podrán ver un listado con todas las asignaturas que tienen.
Precondiciones	<ul style="list-style-type: none">• El usuario se ha logueado en la aplicación.
Postcondiciones	La aplicación muestra una lista con todas las asignaturas.
Flujo principal	<ol style="list-style-type: none">1. Abrir la aplicación.2. Introducir el usuario y la contraseña de la cuenta de Twitter que se desee utilizar.3. Pulsar el botón aceptar.
Historia de usuario	HU-02-A.

Tabla 24. CU-02

Identificador	CU-03
Título	Listar evaluaciones
Actores	Profesor y Alumno
Descripción	Los usuarios de la aplicación podrán ver un listado con todas las evaluaciones disponibles para una determinada asignatura. Además, se mostrará si la evaluación se encuentra abierta, cerrada, evaluada, o no evaluada.
Precondiciones	<ul style="list-style-type: none"> El usuario se ha logueado en la aplicación.
Postcondiciones	La aplicación muestra una lista con todas las evaluaciones.
Flujo principal	<p>Para el profesor:</p> <ol style="list-style-type: none"> 1. Abrir la aplicación. <p>Para el alumno:</p> <ol style="list-style-type: none"> 1. Abrir la aplicación 2. Seleccionar la asignatura de la que se quieren ver las evaluaciones.
Historia de usuario	HU-02-P, HU-03-A, HU-03-P, HU-04-A

Tabla 25. CU-03

Identificador	CU-04
Título	Listar grupos
Actores	Profesor
Descripción	Para cada evaluación, los profesores podrán ver un listado de todos los grupos que forman parte de dicha evaluación. Además se podrá observar que grupos han sido ya evaluados.
Precondiciones	<ul style="list-style-type: none"> El profesor se ha logueado en la aplicación.
Postcondiciones	La aplicación muestra una lista con todos los grupos de alumnos para esa evaluación.
Flujo principal	<ol style="list-style-type: none"> 1. Abrir la aplicación. 2. Seleccionar una evaluación.
Historia de usuario	HU-05-P, HU-06-P.

Tabla 26. CU-04

Identificador	CU-05
Título	Realizar evaluación
Actores	Profesor y Alumno
Descripción	Los usuarios podrán realizar una evaluación en la que valorarán que criterios de la lista de criterios ha cumplido el grupo que se desea evaluar. En el caso de los alumnos, el grupo será el grupo del que ellos forman parte y además podrán evaluar que alumnos de su grupo han colaborado en la realización de cada criterio.
Precondiciones	<ul style="list-style-type: none"> • El usuario se ha logueado en la aplicación. • El usuario aún no ha realizado la evaluación que desea realizar.
Postcondiciones	La evaluación ha sido enviada al servidor en el que se almacenarán todas las evaluaciones realizadas.
Flujo principal	<p>Para el profesor:</p> <ol style="list-style-type: none"> 1. Abrir la aplicación. 2. Seleccionar una evaluación. 3. Seleccionar un grupo de alumnos. 4. Completar el formulario de criterios. 5. Pulsar el botón “Enviar”. <p>Para el alumno:</p> <ol style="list-style-type: none"> 1. Abrir la aplicación. 2. Seleccionar una asignatura. 3. Seleccionar una evaluación. 4. Completar el formulario de criterios. 5. Pulsar el botón “Enviar”.
Historia de usuario	HU-07-P, HU-05-A.

Tabla 27. CU-05

Identificador	CU-06
Título	Ver evaluaciones realizadas
Actores	Profesor y Alumno
Descripción	Para cada evaluación realizada, el usuario podrá ver qué valores fueron introducidos en el momento en el que la evaluación fue realizada.
Precondiciones	<ul style="list-style-type: none"> • El usuario se ha logueado en la aplicación. • El usuario ha realizado al menos una evaluación.
Postcondiciones	La aplicación muestra los valores introducidos cuando se realizó la evaluación.
Flujo principal	<ol style="list-style-type: none"> 1. Abrir la aplicación. 2. Seleccionar una evaluación.
Historia de usuario	HU-08-P, HU-06-A.

Tabla 28. CU-06

Identificador	CU-07
Título	Cerrar evaluación
Actores	Profesor
Descripción	El profesor podrá cerrar una evaluación para que los alumnos no puedan realizarla a partir de ese momento.
Precondiciones	<ul style="list-style-type: none"> • El profesor se ha logueado en la aplicación. • El profesor ha evaluado a todos los grupos de la evaluación que se desea cerrar.
Postcondiciones	La evaluación se muestra como “Cerrada” en la lista de evaluaciones.
Flujo principal	<ol style="list-style-type: none"> 1. Abrir la aplicación. 2. Mantener pulsada una evaluación (saldrá un mensaje de confirmación para cerrar). 3. Confirmar.
Historia de usuario	HU-04-P

Tabla 29. CU-07

Identificador	CU-08
Título	Ver notas evaluaciones.
Actores	Profesor
Descripción	Para cada evaluación, el profesor podrá ver un listado con las notas de todos los grupos que forman parte de esa evaluación.
Precondiciones	<ul style="list-style-type: none"> • El profesor se ha logueado en la aplicación. • El profesor ha cerrado la evaluación de la que desea visualizar las notas.
Postcondiciones	La aplicación muestra una lista con todos los grupos de alumnos para esa evaluación y podrá consultar la nota de cada grupo.
Flujo principal	<ol style="list-style-type: none"> 1. Abrir la aplicación. 2. Seleccionar una evaluación cerrada. 3. Seleccionar el grupo del que desea consultar la nota.
Historia de usuario	HU-10-P

Tabla 30. CU-08

Identificador	CU-09
Título	Ver listado alumnos
Actores	Profesor
Descripción	El profesor podrá ver un listado con todos los alumnos de su asignatura para más tarde consultar sus notas.
Precondiciones	<ul style="list-style-type: none"> • El profesor se ha logueado en la aplicación.
Postcondiciones	La aplicación muestra una lista con todos los alumnos de su asignatura.
Flujo principal	<ol style="list-style-type: none"> 1. Abrir la aplicación. 2. Seleccionar la pestaña “Notas”
Historia de usuario	HU-09-P

Tabla 31. CU-09

Identificador	CU-10
Título	Ver notas alumno.
Actores	Profesor y Alumno
Descripción	Para un alumno dado, el usuario podrá ver las notas obtenidas por dicho alumno en todas las evaluaciones que se encuentren cerradas.
Precondiciones	<ul style="list-style-type: none"> El usuario se ha logueado en la aplicación.
Postcondiciones	La aplicación muestra una lista con la nota obtenida en todas las evaluaciones que se encuentren cerradas.
Flujo principal	<p>Para el profesor:</p> <ol style="list-style-type: none"> 1. Abrir la aplicación. 2. Seleccionar la pestaña “Notas”. 3. Seleccionar el alumno que se desee. <p>Para el alumno:</p> <ol style="list-style-type: none"> 1. Abrir la aplicación. 2. Seleccionar la pestaña “Notas”.
Historia de usuario	HU-09-P, HU-07-A

Tabla 32. CU-10

Identificador	CU-11
Título	Cerrar sesión.
Actores	Profesor y Alumno
Descripción	El usuario podrá cerrar la sesión para iniciar una nueva.
Precondiciones	<ul style="list-style-type: none"> El usuario se ha logueado en la aplicación.
Postcondiciones	La aplicación mostrará la pantalla de login.
Flujo principal	<ol style="list-style-type: none"> 1. Abrir la aplicación. 2. Pulsar el botón menú del dispositivo. 3. Pulsar la opción “Cerrar sesión”
Historia de usuario	HU-11-P, HU-08-A

Tabla 33. CU-11

3.3 Requisitos de restricción

En este apartado hablaremos de los requisitos de restricción que presenta el sistema, es decir, que situaciones deben cumplirse para poder utilizar la aplicación.

Para representar los requisitos de restricción se utilizará el siguiente modelo:

Identificador	
Descripción	

Identificador: Etiqueta que representa a un requisito de restricción de forma única e inequívoca. El identificador presentará el siguiente formato: RR-XX.

Descripción: En este campo se incluirá una descripción del requisito de restricción.

Identificador	RR-01
Descripción	El usuario deberá tener una cuenta en la red social Twitter para poder acceder a la aplicación.

Tabla 34. RR-01

Identificador	RR-02
Descripción	El dispositivo en el que está instalada la aplicación deberá tener acceso a internet para permitir el acceso a la misma.

Tabla 35. RR-02

Identificador	RR-03
Descripción	El servidor web encargado de recibir las peticiones de la aplicación deberá encontrarse en funcionamiento.

Tabla 36. RR-03

Identificador	RR-04
Descripción	El administrador de la asignatura deberá subir una serie de ficheros (lista de alumnos, grupos, criterios de una evaluación...) con un formato específico al servidor en el que se encuentren los datos.

Tabla 37. RR-04

Capítulo IV. Diseño de la aplicación

A lo largo de este capítulo se detallará el diseño de la aplicación a realizar.

Para ello se analizarán en primer lugar cuáles han sido las distintas alternativas de diseño estudiadas y se explicará cual ha sido el motivo de llevar a cabo la alternativa seleccionada.

Tras esto se realizarán las interfaces que mostrarán tanto la aplicación de profesor como la aplicación de alumno.

Finalmente, se mostrarán los diagramas necesarios para la correcta implementación de la aplicación, es decir, los diagramas de clases y los diagramas de secuencia.

4.1 Alternativas de diseño

En esta sección se mostrarán las distintas alternativas de diseño que se han estudiado a la hora de realizar la aplicación.

Las alternativas de diseño estudiadas se dividirán en tres grupos:

- En primer lugar se estudiará cuantas aplicaciones deben realizarse, es decir, si se debe realizar una única aplicación que utilizarán tanto profesores como alumnos, o si se realizará una aplicación distinta para cada uno de ellos.
- En segundo lugar se estudiará cual es la mejor forma de obtener y almacenar los datos necesarios para el correcto funcionamiento de la aplicación, es decir, las asignaturas existentes, los alumnos que contiene cada asignatura, los grupos de alumnos de cada evaluación, las evaluaciones enviadas por alumnos y profesores y las notas obtenidas por cada uno de los alumnos.
- En tercer lugar se estudiará quién será el encargado de proporcionar acceso a los distintos ficheros existentes y el encargado de gestionar la información. Esto podrá ser realizado por la propia aplicación o por un servidor web.

Para cada una de las alternativas estudiadas se detallará cuáles son las ventajas y desventajas que presentan.

4.1.1 Alternativas de diseño: Número de aplicaciones.

En este apartado se estudiarán las ventajas y desventajas de realizar una única aplicación que compartirán tanto profesores como alumnos, o de realizar una aplicación para cada uno de ellos.

4.1.1.1 Alternativa 1. Una única aplicación

De acuerdo con esta alternativa se realizará una única aplicación que sirva tanto para profesores como para alumnos.

Esta alternativa presenta las siguientes ventajas:

- Al compartirse una parte de las interfaces y funcionalidades entre las aplicaciones de profesor y alumno se comparte también una gran parte del código. Este hecho hace que la programación de una única aplicación pueda llevar menos tiempo y por tanto resultar menos costosa que la otra alternativa.
- Si alguno de los profesores desea por algún motivo tener también la aplicación para alumnos, no será necesario tener dos aplicaciones distintas instaladas en su dispositivo móvil.

En cuanto los inconvenientes, son los siguientes:

- El mantenimiento de la misma puede resultar más difícil de llevar a cabo al tener que separar ambas partes del código.
- Cuando se requiera la actualización de la aplicación, tanto profesores como alumnos deberán llevarla a cabo, aunque la actualización únicamente afecte a una de las partes.

4.1.1.2 Alternativa 2. Dos aplicaciones distintas

De acuerdo con esta alternativa se realizará una aplicación para los profesores y otra aplicación para los alumnos.

Las ventajas que presenta esta alternativa son las siguientes:

- Cuando se realice una actualización para una de las aplicaciones únicamente deberán llevarla a cabo los afectados por esa actualización.
- El tamaño cada una de las aplicaciones será menor que si profesores y alumnos compartirán aplicación, ya que no será necesario incluir funcionalidades innecesarias para cada una de las partes.
- El mantenimiento de dichas aplicaciones resultará más sencillo.

Entre sus inconvenientes se encuentran los siguientes:

- El coste de llevar a cabo dos aplicaciones puede resultar más elevado, ya que existe código compartido entre ambas aplicaciones.
- Si un usuario desea tener ambos roles (profesor y alumno) deberá tener instaladas ambas aplicaciones en su terminal.

4.1.1.3 Alternativa escogida

Tras estudiar las ventajas e inconvenientes de cada una de las alternativas expuestas anteriormente se ha decidido llevar a cabo la alternativa dos, es decir, se realizará una aplicación distinta para profesores y para alumnos.

El principal motivo que nos ha llevado a seleccionar esta alternativa ha sido la comodidad del usuario. Gracias a esta alternativa el usuario se ahorrará actualizaciones innecesarias y el tamaño de descarga y de almacenamiento de la aplicación será inferior.

4.1.2 Alternativas de diseño: Almacenamiento de la información.

En este apartado se estudiarán las distintas opciones existentes en cuanto a dónde y cómo se almacenará la información necesaria para el correcto funcionamiento de la aplicación.

Esta información hará referencia a las asignaturas existentes, los alumnos de cada asignatura, las evaluaciones de cada asignatura, los grupos de cada evaluación, las evaluaciones enviadas por los alumnos y profesores y la nota obtenida por cada uno de ellos.

Las distintas alternativas contempladas son las siguientes: almacenamiento de la información en la nube, almacenamiento en un servidor privado o almacenamiento en un servidor FTP alojado en la universidad.

4.1.2.1 Alternativa 1. Almacenamiento en la nube

El almacenamiento en la nube consiste en un modelo de servicio mediante el cual los datos de un sistema se almacenan, se administran y se respaldan de forma remota a través de servidores ajenos y que son administrados por un proveedor de servicios.

Mediante esta alternativa, todos los ficheros e información necesaria se almacenarían y gestionarían a través de la plataforma Dropbox.



Ilustración 10. Alternativa 1. Dropbox

Las ventajas de esta alternativa son las siguientes:

- Al tratarse de un servidor ajeno, no será necesario llevar a cabo el mantenimiento del servidor, ya que será realizado por el propio Dropbox.
- No será necesario contar con espacio de almacenamiento en nuestros servidores.

Por el contrario, esta alternativa cuenta con las siguientes desventajas:

- Al tratarse de un servidor ajeno no es posible estar completamente seguro de que la información no sea modificada. Debido a esto no se puede garantizar la integridad de la información.
- En caso de fallo en el sistema no existe forma de saber cuándo o cómo va a solucionarse. Debido a esto no es posible garantizar la disponibilidad de los datos.

4.1.2.2 Alternativa 2. Almacenamiento en un servidor de la universidad

En esta alternativa de diseño la información se encontraría almacenada en un servidor FTP ya existente proporcionado por la universidad.



Ilustración 11. Alternativa 2. Servidor FTP Universidad

Entre las ventajas de esta alternativa destacan:

- El servidor ya se encuentra en funcionamiento, por lo que no será necesario implementarlo ni ponerlo en marcha.
- La información será gestionada por el administrador del servidor y por los profesores de las asignaturas alojadas en el mismo.
- Se garantiza la integridad, confidencialidad y disponibilidad de los datos.

Los principales inconvenientes son los siguientes:

- El mantenimiento del servidor deberá ser realizado por los responsables de la aplicación y del servidor, incluyendo todos los costes que esto pueda conllevar.

4.1.2.3 Alternativa 3. Almacenamiento en un servidor propio

Mediante esta alternativa se pretende crear un nuevo servidor FTP en el cual almacenar todos los ficheros y la información necesaria. Este servidor estaría dedicado única y exclusivamente a la aplicación que se desea desarrollar.



Ilustración 12. Alternativa 2. Servidor FTP propio

Las principales ventajas de esta alternativa son las siguientes:

- La información será gestionada por el administrador del servidor y por los profesores de las asignaturas alojadas en el mismo.
- Se garantiza la integridad de la información.
- Se garantiza la confidencialidad de la información.

Por el contrario, entre sus principales inconvenientes podemos encontrar los siguientes:

- El servidor deberá ser implementado y puesto en marcha durante el desarrollo del proyecto, lo cual supondrá un incremento en el coste y la duración del desarrollo del mismo.
- El mantenimiento del servidor deberá ser realizado por los responsables de la aplicación y del servidor, incluyendo todos los costes de tiempo o dinero que esto pueda conllevar.

4.1.2.4 Alternativa escogida

La alternativa escogida tras valorar las ventajas e inconvenientes de cada una de las opciones ha sido la alternativa dos, es decir, el almacenamiento en un servidor FTP de la universidad.

Los principales motivos que han llevado a seleccionar esta alternativa han sido su fácil mantenimiento, la seguridad de la información y el hecho de que dicho servidor ya se encontraba en funcionamiento.

4.1.3 Alternativas de diseño: Tratamiento de la información

En este apartado se valorará si es conveniente la creación de un servicio web que tratará y gestionará la información existente o si debe ser la propia aplicación móvil la que se encargue de descargar, leer y subir ficheros al servidor en el que se encuentre almacenada la información.

4.1.3.1 Alternativa 1. Información tratada por la aplicación móvil

De acuerdo con esta alternativa la aplicación móvil será la encargada de realizar todas las operaciones necesarias para el intercambio de información entre el servidor FTP y el dispositivo móvil. Además se encargará también de leer e interpretar toda la información existente.



Ilustración 13. Alternativa 1. Información gestionada por el terminal

Esta alternativa presenta las siguientes ventajas:

- No será necesario la implementación y puesta en funcionamiento de ningún servidor web.
- El funcionamiento de la aplicación no dependerá de que el servicio web se encuentre en funcionamiento.

En cuanto a los inconvenientes, son los siguientes:

- Toda la información y operaciones serán llevadas a cabo por el dispositivo móvil, lo que supondrá un incremento en el tiempo transcurrido desde que se realiza una petición hasta que la respuesta es obtenida.
- Al ser necesario la descarga de toda la información en el dispositivo móvil, se incrementará el tamaño de la información descargada y almacenada en el mismo.
- Al ser necesario incluir un mayor número de funcionalidades en la aplicación, el tamaño de esta se verá incrementado.

4.1.3.2 Alternativa 2. Información tratada por un servidor web

Esta alternativa consiste en la creación de un servidor web que se encargue de realizar todas las operaciones necesarias para dotar de funcionalidad a la aplicación móvil.

De este modo será el servidor web el encargado de descargar ficheros desde el servidor FTP, leer e interpretar dicha información, responder a las peticiones realizadas por la aplicación móvil, realizar los cálculos necesarios para determinar la nota de cada alumno, y subir ficheros al servidor FTP.



Ilustración 14. Alternativa 2. Información gestionada por un servidor web

Las ventajas que presenta esta alternativa son las siguientes:

- Menor tiempo de computo, lo que equivale a una mayor rapidez de la aplicación.
- Menor tamaño de información descargada por el dispositivo móvil.
- Mayor facilidad para la realización de actualizaciones. Además no será necesario que el usuario descargue una nueva actualización cuando el servidor web sea modificado.

Entre sus inconvenientes se encuentran los siguientes:

- Será necesaria la creación y puesta en marcha del servidor web, lo cual supondrá un mayor tiempo y coste en el desarrollo del sistema.

4.1.1.3 Alternativa escogida

Tras estudiar las ventajas y desventajas de cada una de las alternativas expuestas anteriormente se ha decidido llevar a cabo la alternativa dos, es decir, se llevará a cabo la creación de un servidor web encargado de tratar toda la información y de realizar todas las operaciones necesarias.

El principal motivo que nos ha llevado a seleccionar esta alternativa ha sido la comodidad del usuario. Gracias a esta alternativa el usuario se ahorrará un gran número de actualizaciones, el tamaño de la información descargada será menor y la aplicación funcionará con mayor rapidez.

4.2 Paper prototype

En este apartado se mostrarán las distintas interfaces que contiene la aplicación. Gracias a ello podremos tener una idea que de forma rápida y sencilla nos indique que es lo que queremos implementar.

Para llevar a cabo esta tarea se ha utilizado la herramienta FluidUI, que permite la realización fácil, rápida e interactiva de las pantallas que presenta la aplicación. Este software de diseño está disponible a través de la página web www.fluidui.com y permite realizar tanto diseño web como diseño de aplicaciones android o IOS.

Aunque estos serán los esquemas utilizados a la hora de implementar la aplicación y las pantallas en las que se basará la solución final, estas pantallas pueden sufrir ligeras modificaciones durante su desarrollo.

A continuación se mostrarán las interfaces de las aplicaciones tanto de alumnos como de profesores, comenzando por la aplicación de los profesores.

4.2.1 Aplicación de profesor

En este apartado se mostrarán las distintas interfaces que presentará la aplicación para profesores.

En primer lugar mostraremos la interfaz que presentará la aplicación durante el login del usuario:

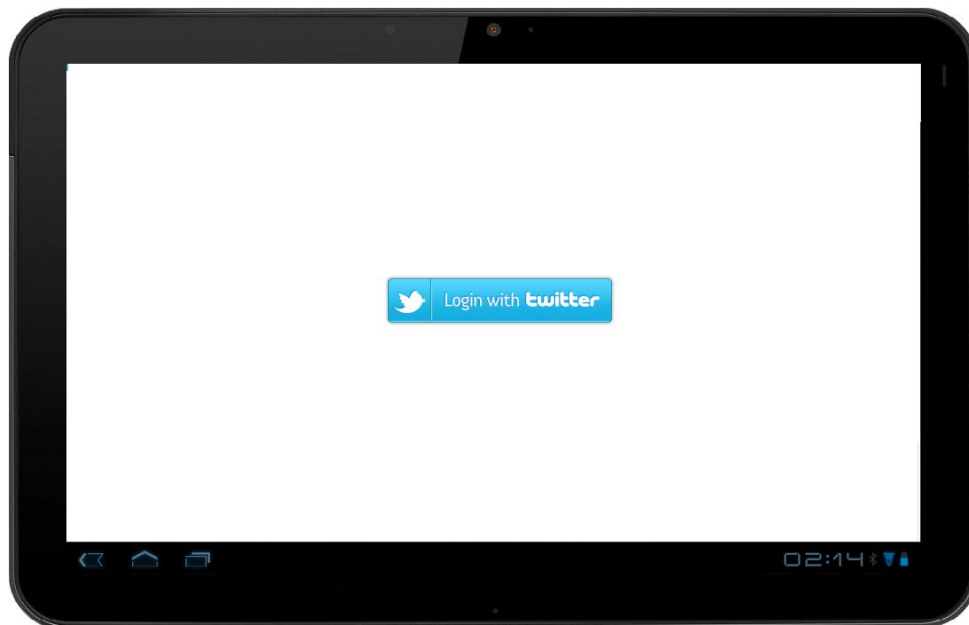


Ilustración 15. Pantalla de login

Tras loguearse en la aplicación se mostrará la pantalla que contendrá el listado de evaluaciones disponibles:

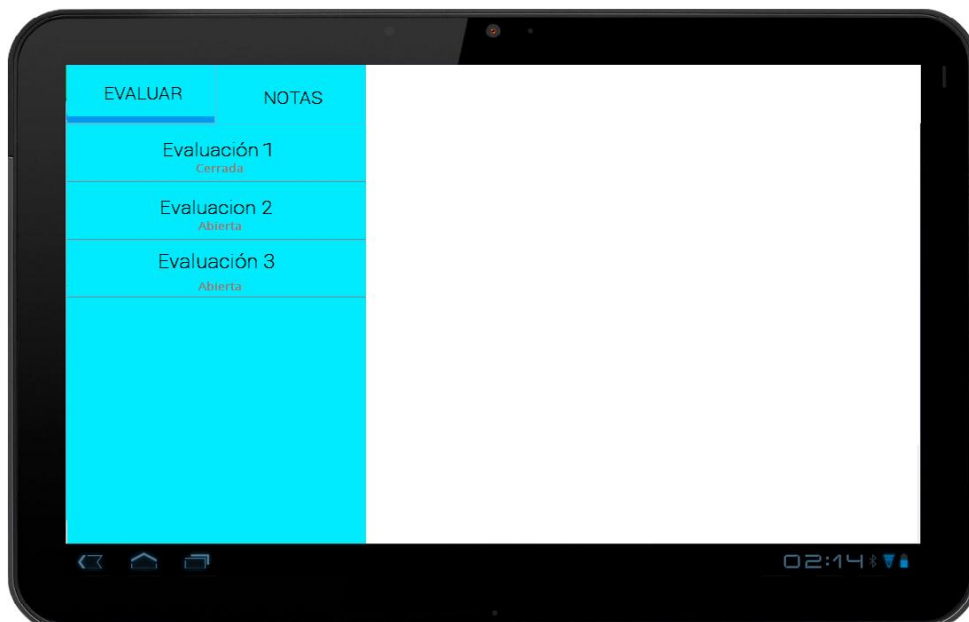


Ilustración 16. Pantalla de listado de evaluaciones

Desde la pantalla que contiene el listado de evaluaciones, al pulsar sobre una evaluación, se mostrará el listado de grupos que contiene dicha evaluación:

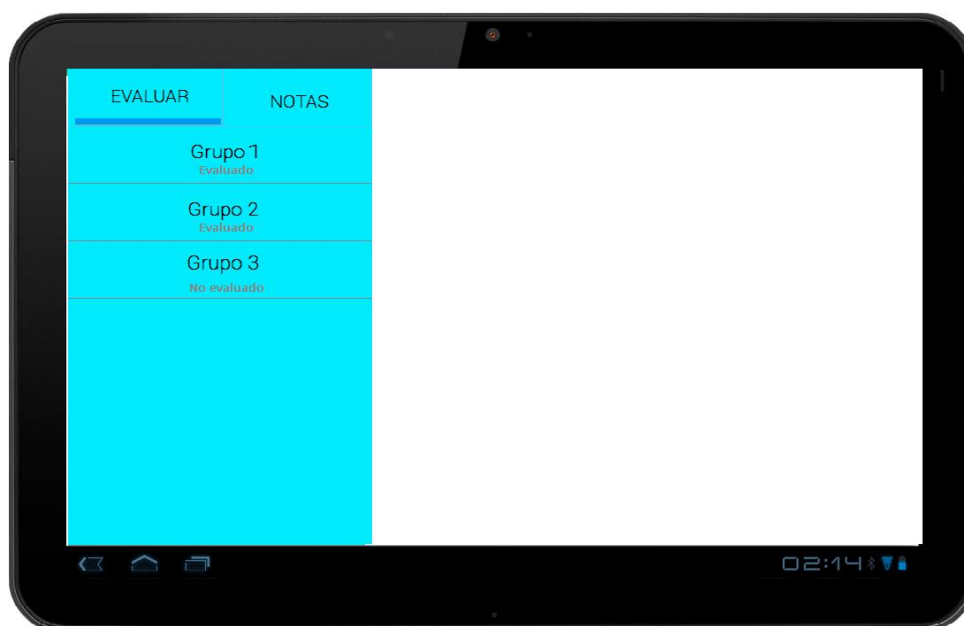


Ilustración 17. Pantalla de listado de grupos

Al pulsar sobre uno de los grupos evaluados, podremos ver la pantalla con la evaluación que el profesor ha realizado previamente:

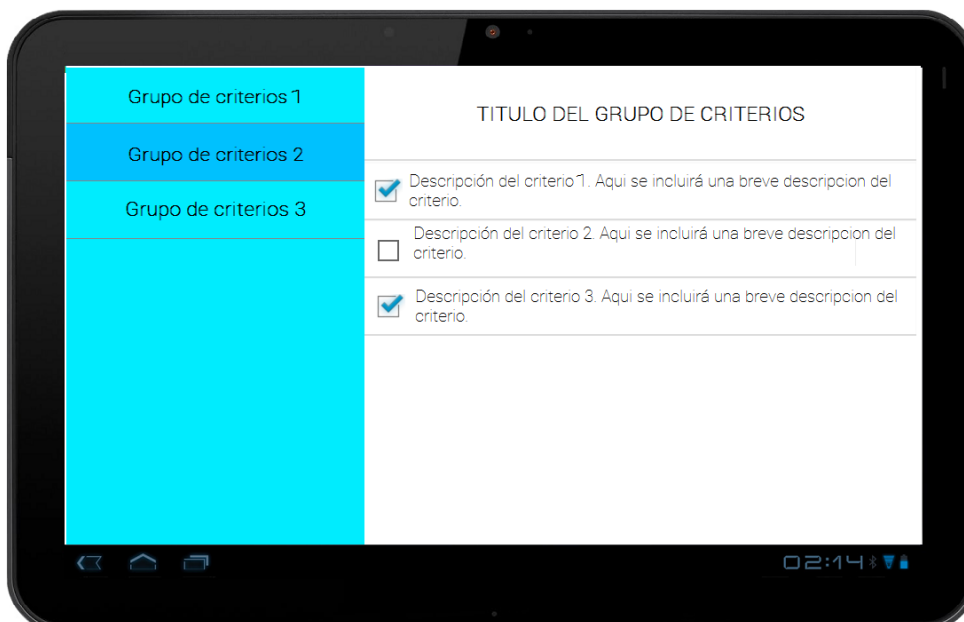


Ilustración 18. Pantalla de visualización de evaluaciones realizadas

Si al pulsar sobre uno de los grupos en la pantalla de listado de grupos este aún no ha sido evaluado veremos la siguiente interfaz:

Grupo de criterios 1	TITULO DEL GRUPO DE CRITERIOS
Grupo de criterios 2	<input checked="" type="checkbox"/> Descripción del criterio 1. Aquí se incluirá una breve descripción del criterio.
Grupo de criterios 3	<input type="checkbox"/> Descripción del criterio 2. Aquí se incluirá una breve descripción del criterio.
	<input checked="" type="checkbox"/> Descripción del criterio 3. Aquí se incluirá una breve descripción del criterio.

Enviar Evaluación

Ilustración 19. Pantalla de evaluación

Si desde la pantalla en la que aparece el listado de evaluaciones pulsamos sobre la pestaña notas podremos ver un listado de todos los alumnos de la asignatura.

EVALUAR	NOTAS
Alumno 1	
Alumno 2	
Alumno 3	

Ilustración 20. Pantalla de listado de alumnos

Al seleccionar un alumno se mostrará el listado de evaluaciones que tienen disponible la nota junto con la nota obtenida por el alumno en la evaluación:

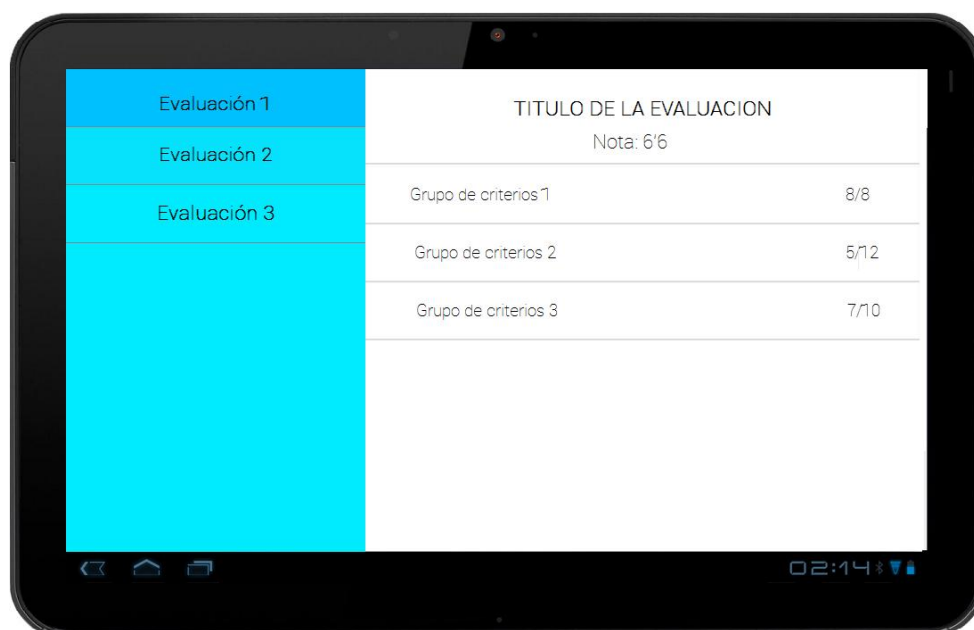


Ilustración 21. Pantalla de notas de alumno

Al seleccionar la pestaña notas desde la interfaz que muestra el listado de grupos se visualizará la siguiente interfaz:

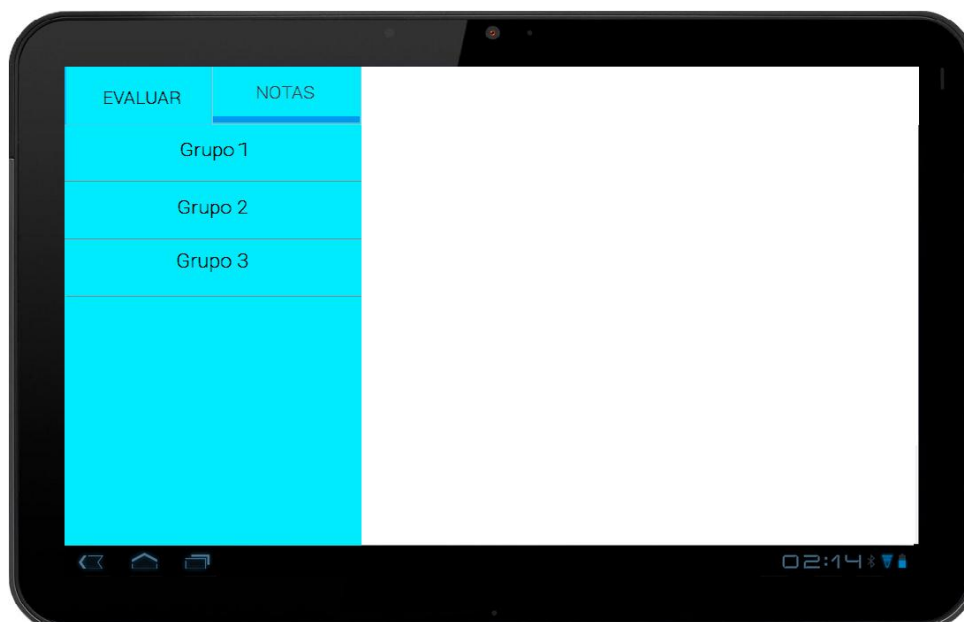


Ilustración 22. Pantalla de listado de grupos (Notas)

Al seleccionar una de los grupos de la interfaz anterior podremos ver la siguiente pantalla:



TITULO DE LA EVALUACION	
Nota: 6/6	
Grupo de criterios 1	8/8
Grupo de criterios 2	5/12
Grupo de criterios 3	7/10

Ilustración 23. Pantalla de notas de grupo

4.2.2 Aplicación de alumno

En primer lugar mostraremos pantalla de login de la aplicación del alumno:

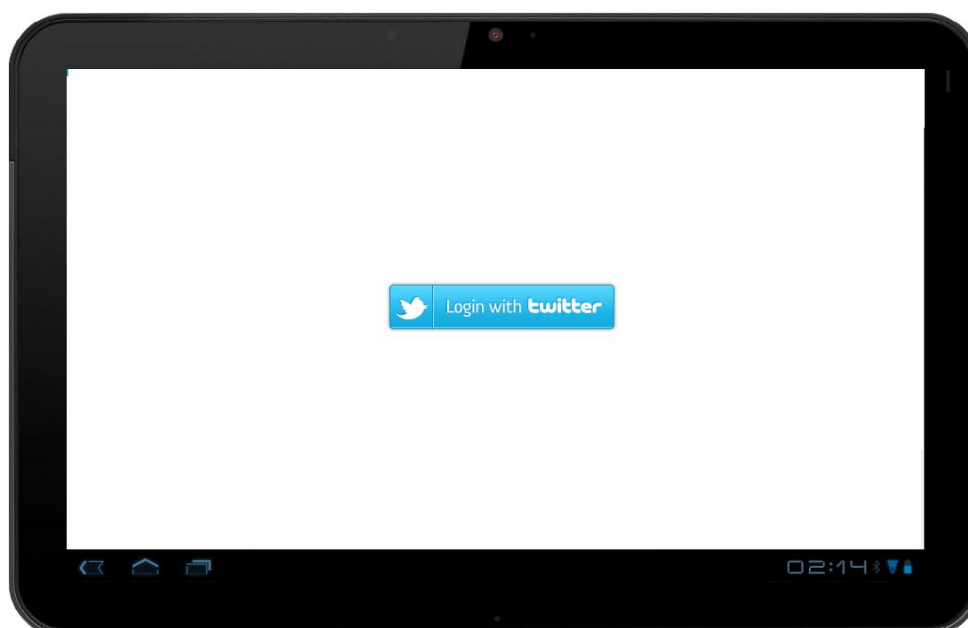


Ilustración 24. Pantalla de login

Tras loguearse en la aplicación se mostrará la interfaz que contiene el listado de asignaturas del alumno:

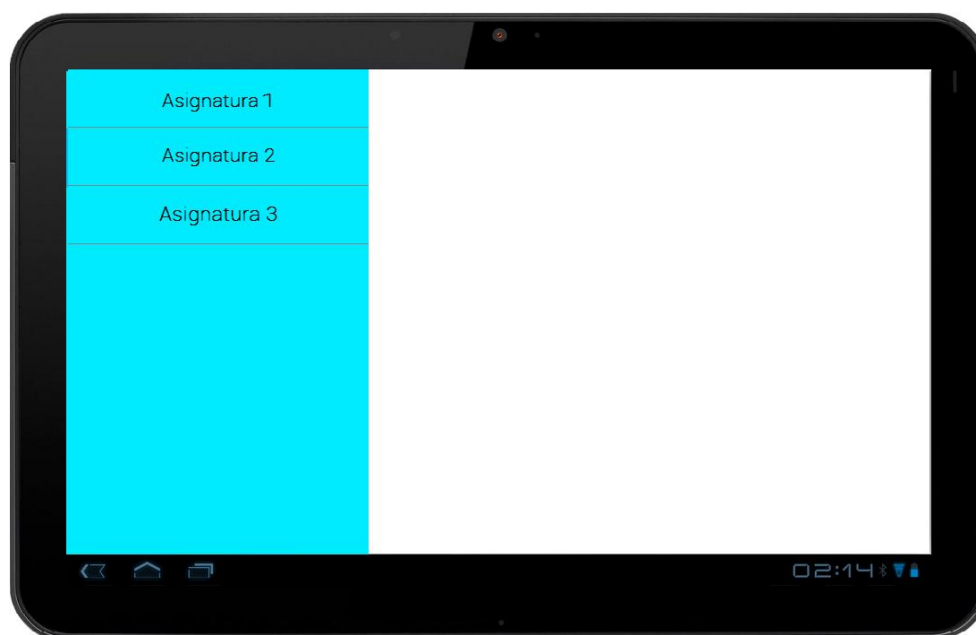


Ilustración 25. Pantalla de listado de asignaturas

Al seleccionar una asignatura se mostrará el listado de evaluaciones de dicha asignatura:

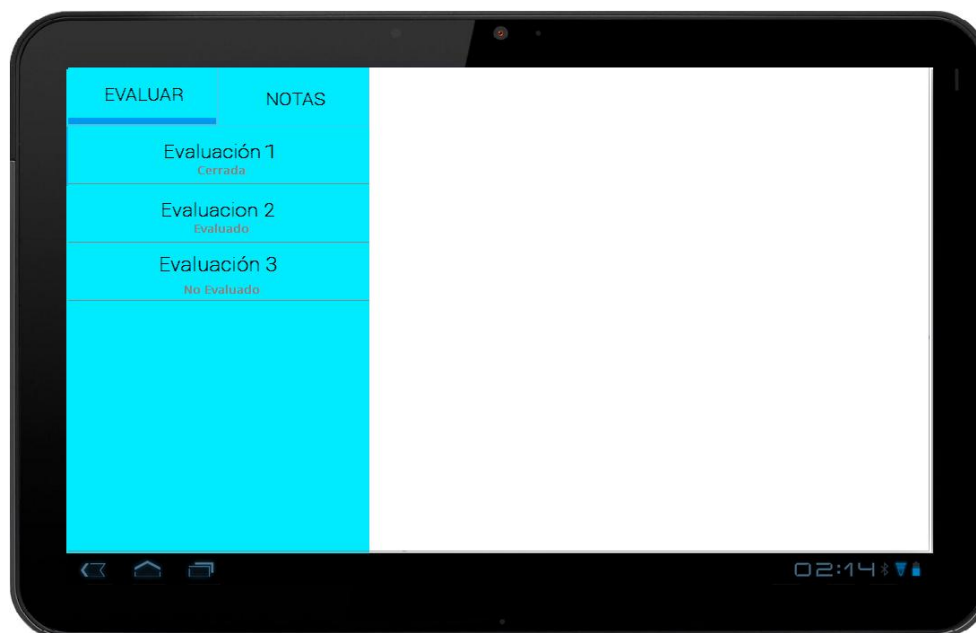


Ilustración 26. Pantalla de listado de evaluaciones

Al seleccionar una evaluación cerrada o ya realizada se mostrará la siguiente interfaz, que permitirá visualizar la evaluación realizada:



Ilustración 27. Pantalla de visualización de evaluaciones realizadas

Si la evaluación seleccionada aún no ha sido realizada se mostrará la siguiente interfaz:

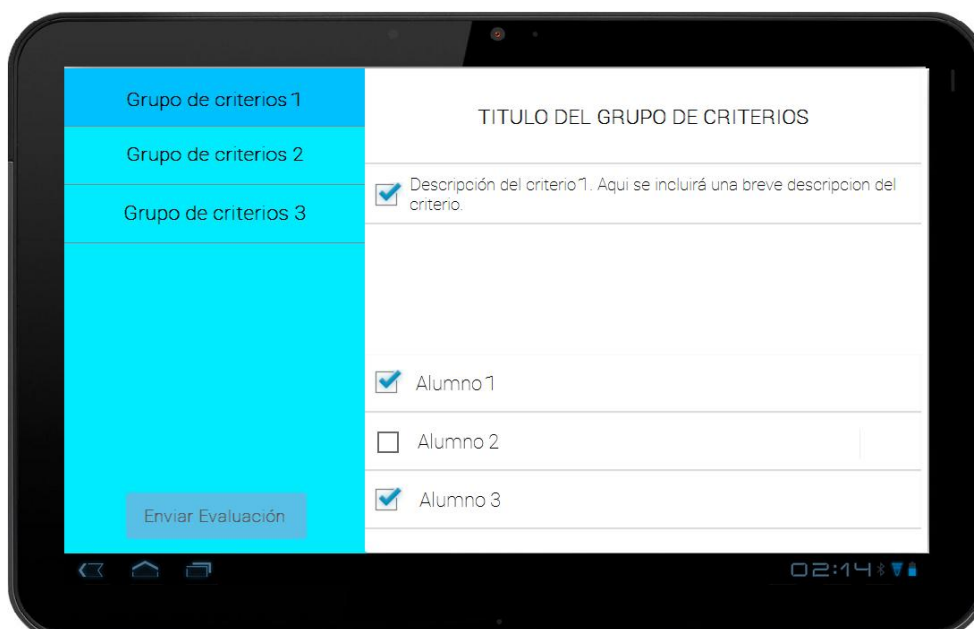


Ilustración 28. Pantalla de evaluaciones

Si desde la pantalla que muestra el listado de evaluaciones pulsamos en la pestaña de notas se mostrarán las evaluaciones que tienen una nota disponible:

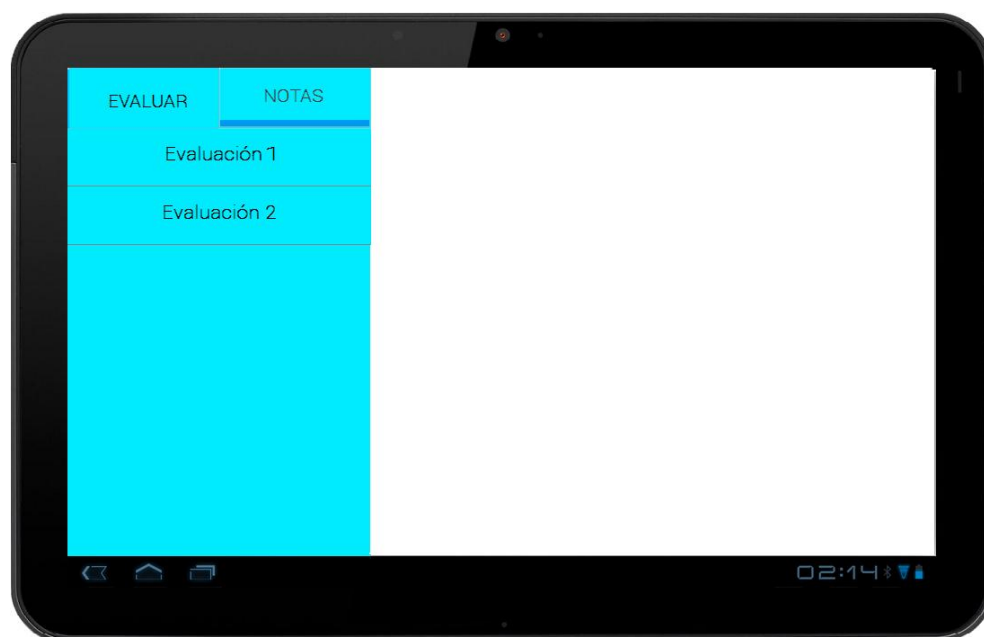


Ilustración 29. Pantalla de listado de notas

Al seleccionar una evaluación se mostrará la nota obtenida en esa evaluación tanto por el alumno, como por el grupo de alumnos:



Ilustración 30. Pantalla de notas

4.3 Diagrama de clases

Un diagrama de clases es un tipo de diagrama estático que muestra las diferentes clases que forman un sistema y las relaciones existentes entre dichas clases.

Aunque las aplicaciones de alumno y profesor contendrán clases muy similares, es necesario realizar un diagrama de clases para cada una de ellas debido a las pequeñas diferencias que puedan existir entre ellas.

En cuanto al servicio web será implementado mediante Play Framework, por lo que constará de una única clase que contendrá todos los métodos que responderán a las peticiones realizadas por la aplicación.

En primer lugar comenzaremos con el diagrama de clases de la aplicación para profesores.

4.3.1 Diagrama de clases. Aplicación de profesores

El diagrama de clases correspondiente a la aplicación de profesores es el siguiente:

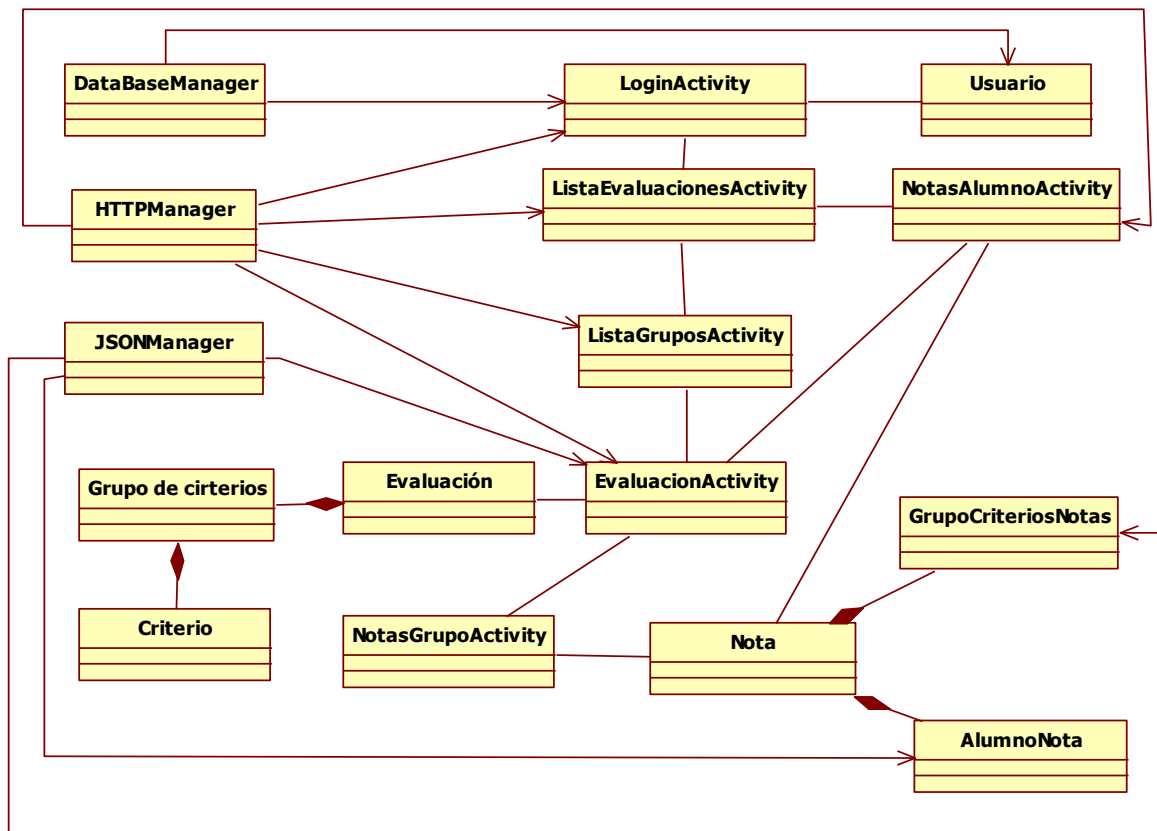


Ilustración 31. Diagrama de clases. Aplicación profesor

A continuación se detallarán los métodos y atributos de cada clase:

Nombre	Usuario
Descripción	Clase que instanciará a los distintos usuarios que acceden a la aplicación.
Atributos	<ul style="list-style-type: none"> • id: atributo de tipo long que almacenará el identificador devuelto por Twitter para el usuario con el que se ha realizado el login • screenName: atributo de tipo String que contendrá el nombre identificativo utilizado en Twitter. Este atributo siempre comenzará por @. • token: atributo de tipo String que contendrá el token devuelto por twitter al realizar el login de un usuario. • tokenSecret: atributo de tipo String que contendrá el tokenSecret devuelto por twitter al realizar el login de un usuario.
Métodos	-

Tabla 38. Clase Usuario.

Nombre	Criterio
Descripción	Clase que instanciará los distintos criterios de una evaluación
Atributos	<ul style="list-style-type: none"> • criterio: atributo de tipo String que contendrá la descripción del criterio. • valor: atributo de tipo int que contendrá el valor del criterio. • seleccionado: atributo de tipo boolean que indicará si el criterio ha sido o no cumplido por el grupo de alumnos.
Métodos	-

Tabla 39. Clase Criterio

Nombre	GrupoCriterios
Descripción	Clase que instanciará los distintos grupos de criterios de una evaluación
Atributos	<ul style="list-style-type: none"> • titulo: titulo del grupo de criterios. • valor: valor del grupo criterios. • valorObtenido: nota que se ha obtenido en el grupo de criterios. • criterios: conjunto de criterios de los que se compone el grupo de criterios.
Métodos	<ul style="list-style-type: none"> • numCriterios(): método que devuelve el número de criterios del grupo de criterios.

Tabla 40. Clase GrupoCriterios

Nombre	Evaluacion
Descripción	Clase que instanciará una determinada evaluación.
Atributos	<ul style="list-style-type: none"> • grupoCriterios: Array que contendrá todos los grupos de criterios que forman una evaluación.
Métodos	<ul style="list-style-type: none"> • numGrupos(): método que devuelve el número de grupos de criterios que contiene la evaluación.

Tabla 41. Clase Evaluacion

Nombre	AlumnoNota
Descripción	Clase que instanciará la nota obtenida por un alumno.
Atributos	<ul style="list-style-type: none"> • nombre: nombre del alumno. • puntosObtenidos: puntos obtenidos por el alumno. • nota: nota obtenida por el alumno.
Métodos	-

Tabla 42. Clase AlumnoNota

Nombre	GrupoCriteriosNota
Descripción	Clase que instanciará la nota obtenida por un grupo de alumnos en un determinado grupo de criterios.
Atributos	<ul style="list-style-type: none"> • titulo: titulo del grupo de criterios. • valor: valor del grupo de criterios. • notaProfesor: nota otorgada por el profesor en el grupo de criterios. • notaAlumnos: media de la nota otorgada por el conjunto de alumnos que conforman el grupo para este grupo de criterios. • notaGrupoCriterios: nota obtenida en el grupo de criterios teniendo en cuenta la nota del profesor y la de los alumnos. • alumnos: array de tipo AlumnoNotas que contendrá a los distintos alumnos del grupo.
Métodos	-

Tabla 43. Clase GrupoCriteriosNota

Nombre	Nota
Descripción	Clase que contiene la nota obtenida por un determinado grupo de alumnos y para una determinada evaluación.
Atributos	<ul style="list-style-type: none"> • grupoCriterios: array de tipo GrupoCriteriosNota. • numAlumnos: número de alumnos que forman el grupo evaluado. • notaMáxima: mayor cantidad de puntos que pueden obtenerse en la evaluación. • nota: nota obtenida por el grupo en la evaluación. • alumnos: array de tipo AlumnoNota que contendrá a todos los alumnos que forman parte del grupo evaluado.
Métodos	-

Tabla 44. Clase Nota

Nombre	HTTPManager
Descripción	Clase encargada de la comunicación con el servidor web. Realizará todas las peticiones necesarias para el correcto funcionamiento de la aplicación. Todos los métodos contenidos en esta clase harán uso del web service para devolver los valores solicitados por la aplicación.
Atributos	<ul style="list-style-type: none"> • url: Atributo de tipo String que contendrá la dirección del web service.
Métodos	<ul style="list-style-type: none"> • serverIsOn(): método que realizará una petición al web service para determinar si este se encuentra en funcionamiento. • tipoCuenta(String usuario): método que determinará si el usuario que se ha introducido durante el login se corresponde con una cuenta de profesor o una cuenta de alumno. • listarEvaluaciones(String asignatura): método que devolverá un array que contendrá todas las evaluaciones existentes para la asignatura recibida como parámetro. • listarAlumnos(String asignatura): método que devolverá un array que contendrá un listado con todos los alumnos de la asignatura recibida como parámetro. • evaluaciónRealizada(String asignatura, String evaluación): método que determinará si la evaluación seleccionada ha sido realizada previamente por el profesor. • evaluaciónCerrada(String asignatura, String evaluación): método que determinará si la evaluación seleccionada se encuentra abierta o cerrada. • cerrarEvaluación(String asignatura, String evaluación): método utilizado para cerrar una evaluación para que no

	<p>pueda ser evaluada por nadie más.</p> <ul style="list-style-type: none"> • descargarCriterios(String asignatura, String evaluación): método que descargará del servidor FTP el listado de criterios que contiene una evaluación determinada y devolverá el fichero descargado con formato String. • enviarEvaluación(String asignatura, String evaluación, JSONObject respuesta): método que a través del web service enviará al servidor FTP la evaluación realizada por el profesor. • descargarEvaluación(String asignatura, String evaluación, String grupo): método que devolverá en formato Json la evaluación realizada anteriormente por el profesor para la evaluación y el grupo recibidos como parámetros. • listarGrupos(String asignatura, String evaluación): método que devolverá todos los grupos de alumnos existentes para una evaluación determinada. • obtenerGrupo(String asignatura, String evaluación, String alumno): método que determinará el grupo al que pertenece un alumno dado. • listarNotas(String asignatura): método que devolverá que evaluaciones de la asignatura recibida como parámetro tienen ya la nota disponible. • descargarNotas(String asignatura, String evaluación, String grupo): método que devolverá en formato JSON la nota obtenida por un grupo de alumnos para la evaluación y la asignatura recibidas como parámetros.
--	--

Tabla 45. Clase HTTPManager

Nombre	JSONManager
Descripción	Clase utilizada para gestionar los ficheros JSON recibidos desde el servidor.
Atributos	-
Métodos	<ul style="list-style-type: none"> • leerJSONAlumnos(JSON alumnos): método que devolverá un array de usuarios a partir de un fichero JSON. • leerJSONCriterios(JSON criterios): método que devolverá un objeto Evaluación que contendrá el conjunto de criterios que forman una evaluación. • crearJSONEvaluación(Evaluación eval): método que creará un fichero JSON a partir de la evaluación realizada por el profesor. • leerJSONEvaluación(JSON evaluación): método que devolverá un objeto evaluación a partir de un fichero json. • leerJSONNota(JSON notas): método que devolverá un objeto Nota a partir de un fichero JSON.

Tabla 46. Clase JSONManager

Nombre	DataBaseManager
Descripción	Clase que gestionará la base de datos que contendrá los distintos usuarios logueados en la aplicación.
Atributos	-
Métodos	<ul style="list-style-type: none"> • addUser(Usuario user): método que insertará un usuario en la base de datos. • getAllUsers(): método que devolverá todos los usuario logueados anteriormente en la aplicación. • getUserById(String userId): método que devolverá todos los datos del usuario recibido como parámetro.

Tabla 47. Clase DataBaseManager

Nombre	LoginActivity
Descripción	Clase utilizada para visualizar y llevar a cabo el login en la aplicación.
Atributos	<ul style="list-style-type: none"> • twitter: atributo de la clase twitter4j.Twitter en el que se instanciará el objeto que iniciará sesión en Twitter. • requestToken: atributo de la clase twitter4j.RequestToken en el que se almacenará la respuesta obtenida a través de twitter tras la autenticación • tipoCuenta: atributo de tipo String en el que se almacenará el tipo de cuenta (profesor o alumno) del usuario logueado.
Métodos	<ul style="list-style-type: none"> • conexionDetector(): método que determinará si el dispositivo cuenta con conexión a internet. • isAlreadyLogged(): método que comprobará si hay algún usuario logueado anteriormente en la aplicación. • abrirLoginTwitter(): método que redirigirá la aplicación a la página de autenticación de twitter. • loginApplication(String usuario, String password): método que logueará al usuario en la aplicación.

Tabla 48. Clase LoginActivity.

Nombre	ListadoEvaluacionesActivity
Descripción	Clase que servirá para la visualización de la pantalla que contendrá los alumnos y las evaluaciones disponibles.
Atributos	<ul style="list-style-type: none"> • asignatura: atributo que indicará la asignatura.
Métodos	<ul style="list-style-type: none"> • listarEvaluaciones(): método que devolverá un listado con todas las evaluaciones disponibles. • listarAlumnos(): método que devolverá un listado de todos los alumnos de la asignatura.

Tabla 49. Clase ListadoEvaluacionesActivity

Nombre	ListadoGruposActivity
Descripción	Clase que mostrará los distintos grupos de alumnos para su posterior evaluación.
Atributos	<ul style="list-style-type: none"> • asignatura: atributo que contendrá la asignatura. • evaluación: atributo que contendrá la evaluación seleccionada. • grupos: array que contendrá los grupos existentes.
Métodos	<ul style="list-style-type: none"> • listarGrupos(String asignatura, String evaluación): método que devolverá los grupos de alumnos existentes en una determinada evaluación.

Tabla 50. Clase ListadoGruposActivity

Nombre	EvaluaciónActivity
Descripción	Clase que creará la interfaz de una evaluación y permitirá a los profesores tanto visualizarla como realizarla.
Atributos	<ul style="list-style-type: none"> • asignatura: atributo que contendrá la asignatura. • evaluación: atributo que contendrá la evaluación seleccionada. • grupo: atributo que contendrá el grupo evaluado. • evaluaciónDatos: atributo de tipo Evaluación que contendrá los datos que el profesor ha introducido durante la evaluación.
Métodos	<ul style="list-style-type: none"> • guardarDatos(): método encargado de almacenar los datos introducidos por el profesor. • enviarEvaluacion(String asignatura, String evaluación, JSON eval): método encargado de enviar la evaluación al servidor. • listarCriterios(String asignatura, String evaluacion): método que mostrará los criterios existentes en una evaluación.

Tabla 51. Clase EvaluacionActivity

Nombre	NotasAlumnoActivity
Descripción	Clase que mostrará todas las notas que un alumno ha obtenido en cada una de las evaluaciones.
Atributos	<ul style="list-style-type: none"> • alumno: atributo que contendrá el nombre del alumno. • asignatura: atributo que contendrá el nombre de la asignatura.
Métodos	<ul style="list-style-type: none"> • listarNotas(String asignatura, String alumno): método que devolverá todas las notas disponibles para un alumno determinado. • pintarNota(String alumno, String asignatura, String evaluación): método que pintará en pantalla la información que contendrá la nota obtenida por el alumno.

Tabla 52. Clase NotasAlumnoActivity

Nombre	NotasGrupoActivity
Descripción	Clase que mostrará la nota obtenida en una evaluación por cada uno de los integrantes de un grupo de alumnos.
Atributos	<ul style="list-style-type: none"> • asignatura: atributo que contendrá la asignatura. • evaluación: atributo que contendrá la evaluación seleccionada. • grupo: atributo que contendrá el grupo visualizado.
Métodos	<ul style="list-style-type: none"> • listarAlumnos(String asignatura, String evaluación, String grupo): método que devolverá los alumnos pertenecientes al grupo seleccionado. • pintarNota(String alumno, String asignatura, String evaluación): método que pintará en pantalla la información que contiene la nota obtenida por el alumno.

Tabla 53. Clase NotasGrupoActivity

4.3.2 Diagrama de clases. Aplicación de alumnos.

El diagrama de clases correspondiente a la aplicación de alumnos es el siguiente:

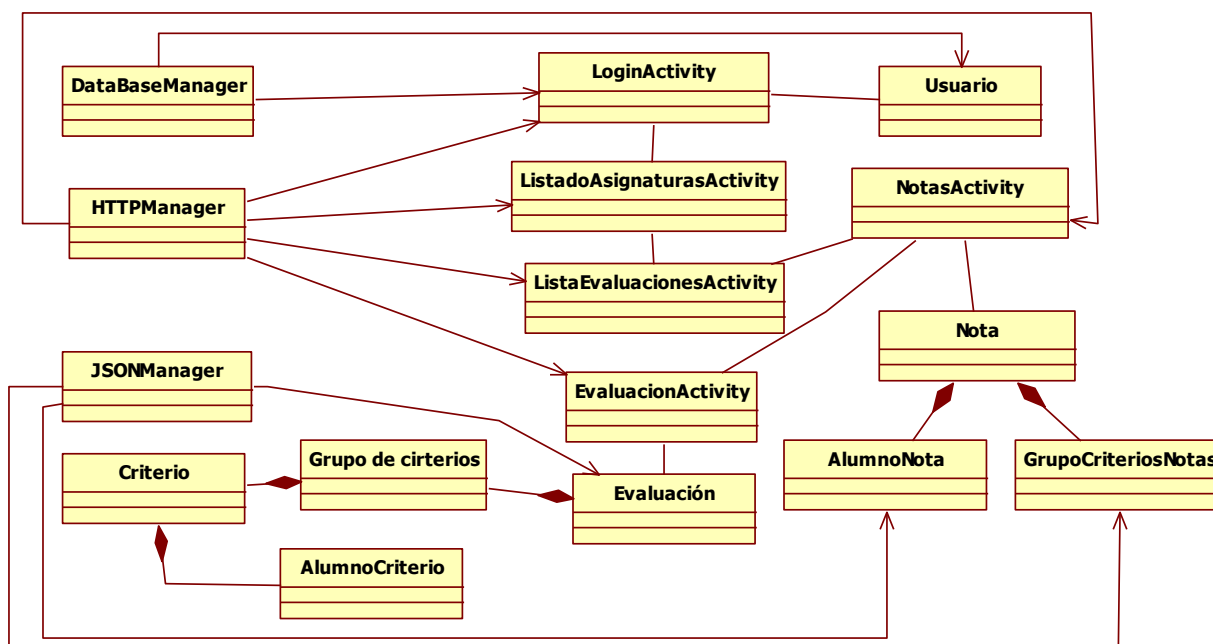


Ilustración 32. Diagrama de clases. Aplicación Alumno

A continuación se detallarán los métodos y atributos de cada clase:

Nombre	Usuario
Descripción	Clase que contendrá los distintos usuarios de la aplicación.
Atributos	<ul style="list-style-type: none"> • id: atributo de tipo long que almacenará el identificador devuelto por Twitter para el usuario. • screenName: atributo de tipo String que contendrá el nombre identificativo utilizado en Twitter. Este atributo siempre comenzará por @. • token: atributo de tipo String que contendrá el token devuelto por twitter al realizar el login de un usuario. • tokenSecret: atributo de tipo String que contendrá el tokenSecret devuelto por twitter al realizar el login de un usuario.
Métodos	-

Tabla 54. Clase Usuario.

Nombre	AlumnoCriterio
Descripción	Clase que representará si un alumno ha trabajado o no en el cumplimiento de un criterio.
Atributos	<ul style="list-style-type: none"> • nombre: nombre del alumno. • seleccionado: atributo de tipo boolean que indicará si el criterio ha sido o no cumplido por el alumno.
Métodos	-

Tabla 55. Clase AlumnoCriterio

Nombre	Criterio
Descripción	Clase que instanciará los distintos criterios de una evaluación.
Atributos	<ul style="list-style-type: none"> • criterio: atributo que contendrá la descripción del criterio. • valor: atributo que contendrá el valor del criterio. • seleccionado: atributo de tipo boolean que indicará si el criterio ha sido o no cumplido por el grupo. • alumnos: array de tipo AlumnoCriterio que contendrá a los alumnos que forman un grupo.
Métodos	-

Tabla 56. Clase Criterio

Nombre	GrupoCriterios
Descripción	Clase que instanciará los distintos grupos de criterios.
Atributos	<ul style="list-style-type: none"> • titulo: titulo del grupo de criterios. • valor: valor del grupo criterios. • valorObtenido: valor que ha obtenido el grupo en el grupo de criterios. • criterios: conjunto de criterios de los que se compone el grupo de criterios.
Métodos	<ul style="list-style-type: none"> • numCriterios(): método que devuelve el número de criterios.

Tabla 57. Clase GrupoCriterios

Nombre	Evaluacion
Descripción	Clase que instanciará una determinada evaluación.
Atributos	<ul style="list-style-type: none"> • grupoCriterios: Array que contendrá todos los grupos de criterios que forman una evaluación.
Métodos	<ul style="list-style-type: none"> • numGrupos(): método que devuelve el número de grupos de criterios que contiene la evaluación.

Tabla 58. Clase Evaluacion

Nombre	AlumnoNota
Descripción	Clase que instanciará la nota obtenida por un alumno.
Atributos	<ul style="list-style-type: none"> • nombre: nombre del alumno. • puntosObtenidos: puntos obtenidos por el alumno. • nota: nota obtenida por el alumno.
Métodos	-

Tabla 59. Clase AlumnoNota

Nombre	GrupoCriteriosNota
Descripción	Clase que instanciará la nota obtenida por un grupo de alumnos en un determinado grupo de criterios.
Atributos	<ul style="list-style-type: none"> • titulo: titulo del grupo de criterios. • valor: valor del grupo de criterios. • notaprofesor: nota otorgada por el profesor. • notaAlumnos: media de la nota otorgada por el conjunto de alumnos que conforman el grupo. • notaGrupoCriterios: nota obtenida en el grupo de criterios teniendo en cuenta la nota del profesor y la de los alumnos. • alumnos: array de tipo AlumnoNotas que contendrá los distintos alumnos del grupo.
Métodos	-

Tabla 60. Clase GrupoCriteriosNota

Nombre	Nota
Descripción	Clase que contiene la nota obtenida por un determinado grupo de alumnos y para una determinada evaluación.
Atributos	<ul style="list-style-type: none"> • grupoCriterios: array de tipo GrupoCriteriosNota. • numAlumnos: número de alumnos que forman el grupo evaluado. • notaMáxima: mayor cantidad de puntos que pueden obtenerse en la evaluación. • nota: nota obtenida por el grupo en la evaluación. • alumnos: array de tipo AlumnoNota que contendrá a todos los alumnos que forman el grupo evaluado.
Métodos	-

Tabla 61. Clase Nota

Nombre	HTTPManager
Descripción	Clase encargada de la comunicación con el servidor web. Realizará todas las peticiones necesarias para el correcto funcionamiento de la aplicación.
Atributos	<ul style="list-style-type: none"> • url: Atributo de tipo String que contendrá la dirección del web service.
Métodos	<ul style="list-style-type: none"> • serverIsOn(): método que determinará si es posible establecer conexión con el servidor web. • tipoCuenta(String usuario): método que determinará si la cuenta utilizada es de profesor o de alumno. • listarAsignaturas(String usuario): devuelve un listado con todas las asignaturas del alumno. • listarEvaluaciones(String asignatura): método que devolverá las evaluaciones existentes para una asignatura determinada. • evaluaciónRealizada(String asignatura, String evaluación): método que determinará si la evaluación

	<p>seleccionada ha sido realizada previamente por el alumno.</p> <ul style="list-style-type: none"> • evaluaciónCerrada(String asignatura, String evaluación): método que determinará si la evaluación seleccionada se encuentra abierta o cerrada. • descargarCriterios(String asignatura, String evaluación): método que descargará del servidor FTP el listado de criterios que contiene una evaluación determinada. • enviarEvaluación(String asignatura, String evaluación, JSONObject respuesta): método que enviará al servidor FTP la evaluación realizada por el alumno en formato JSON. • descargarEvaluación(String asignatura, String evaluación, String grupo): método que devolverá en formato JSON la evaluación realizada anteriormente por el alumno para la evaluación recibida como parámetro para que pueda ser visualizada. • obtenerGrupo(String asignatura, String evaluación, String alumno): método que determinará el grupo al que pertenece un alumno dado. • miembrosGrupo(String asignatura, String evaluación, String alumno): método que devuelve los integrantes de un grupo. • listarNotas(String asignatura): método que devolverá el nombre de las evaluaciones que ya tienen la nota disponible. • descargarNotas(String asignatura, String evaluación, String grupo): método que devolverá en formato JSON la nota obtenida por un grupo de alumnos para una evaluación dada.
--	--

Tabla 62. Clase HTTPManager

Nombre	JSONManager
Descripción	Clase utilizada para gestionar los ficheros JSON recibidos desde el servidor.
Atributos	-
Métodos	<ul style="list-style-type: none"> • leerJSONAlumnos(JSON alumnos): método que devolverá un array de usuarios a partir de un fichero JSON. • leerJSONCriterios(JSON criterios): método que devolverá un objeto Evaluación que contendrá el conjunto de criterios que forman una evaluación. • crearJSONEvaluación(Evaluación eval): método que creará un fichero JSON a partir de la evaluación realizada por el alumno. • leerJSONEvaluación(JSON evaluación): método que transformará un fichero JSON en un objeto Evaluación. • leerJSONNota(JSON notas): método que devolverá un objeto Nota a partir de un fichero JSON.

Tabla 63. Clase JSONManager

Nombre	DataBaseManager
Descripción	Clase que gestionará la base de datos que contendrá los distintos usuarios logueados en la aplicación.
Atributos	-
Métodos	<ul style="list-style-type: none"> • addUser(Usuario user): método que insertará un usuario en la base de datos. • getAllUsers(): método que devolverá todos los usuario logueados anteriormente en la aplicación. • getUserById(String userId): método que devolverá todos los datos de un usuario determinado logueado en la aplicación.

Tabla 64. Clase DataBaseManager

Nombre	LoginActivity
Descripción	Clase utilizada para visualizar y llevar a cabo el login en la aplicación.
Atributos	<ul style="list-style-type: none"> • twitter: atributo de la clase twitter4j.Twitter en el que se instanciará el objeto que se iniciará sesión en Twitter. • requestToken: atributo de la clase twitter4j.RequestToken en el que se almacenará la respuesta obtenida a través de twitter tras la autenticación. • tipoCuenta: atributo de tipo String en el que se almacenará el tipo de cuenta (profesor o alumno) del usuario logueado.
Métodos	<ul style="list-style-type: none"> • conexionDetector(): método que determinará si el dispositivo cuenta con conexión a internet. • isAlreadyLogged(): método que comprobará si hay algún usuario logueado anteriormente en la aplicación. • AbrirLoginTwitter(): método redirigirá la aplicación a la página de autenticación de twitter. • loginApplication(String usuario, String password): método que logueará al usuario en la aplicación.

Tabla 65. Clase LoginActivity.

Nombre	ListadoEvaluacionesActivity
Descripción	Clase que servirá para la visualización de la pantalla que contendrá los alumnos y las evaluaciones disponibles.
Atributos	<ul style="list-style-type: none"> • asignatura: atributo que indicará la asignatura.
Métodos	<ul style="list-style-type: none"> • listarEvaluaciones(): método que devolverá un listado con todas las evaluaciones disponibles. • listarNotas(String asignatura, String usuario): método que devolverá las notas disponibles.

Tabla 66. Clase ListadoEvaluacionesActivity

Nombre	ListadoAsignaturasActivity
Descripción	Clase que mostrará por pantalla las asignaturas disponibles para el usuario.
Atributos	<ul style="list-style-type: none"> • usuario: atributo que contendrá el nombre del usuario con el que se ha realizado el login.
Métodos	<ul style="list-style-type: none"> • listarAsignaturas(String asignatura, String evaluación): método que devolverá los grupos de alumnos existentes en una determinada evaluación.

Tabla 67. Clase ListadoGruposActivity

Nombre	EvaluaciónActivity
Descripción	Clase que creará la interfaz de una evaluación y permitirá a los alumnos tanto visualizarla como realizarla.
Atributos	<ul style="list-style-type: none"> • asignatura: atributo que contendrá la asignatura. • evaluación: atributo que contendrá la evaluación seleccionada. • grupo: atributo que contendrá el grupo que se está evaluando. • evaluaciónDatos: atributo de tipo evaluación que contendrá los datos que el alumno ha introducido durante la evaluación.
Métodos	<ul style="list-style-type: none"> • guardarDatos(): método encargado de almacenar los datos introducidos por el alumno. • enviarEvaluacion(String asignatura, String evaluación, JSON eval): método que envía la evaluación al servidor. • descargarEvaluacion(String asignatura, String evaluación, grupo): método encargado de descargar una evaluación realizada anteriormente. • listarCriterios(String asignatura, String evaluacion): muestra los criterios existentes en una evaluación.

Tabla 68. Clase EvaluacionActivity

Nombre	NotasActivity
Descripción	Clase que mostrará todas las notas que un alumno ha obtenido en cada una de las evaluaciones.
Atributos	<ul style="list-style-type: none"> • usuario: atributo que contendrá el nombre del usuario. • asignatura: atributo que contendrá el nombre de la asignatura. • evaluación: atributo que contendrá la evaluación de la que se está visualizando la nota.
Métodos	<ul style="list-style-type: none"> • listarMiembros(String asignatura, String alumno): método que mostrará un listado que permitirá seleccionar entre la nota del alumno o la nota del grupo. • pintarNota(String alumno, String asignatura, String evaluación): método que pintará en pantalla la información que contendrá la nota obtenida por el alumno.

Tabla 69. Clase NotasAlumnoActivity

4.3.3 Diagrama de clases. Servicio Web.

El servicio web será realizado a través del framework PlayFramework. A través de este framework todas las peticiones serán tratadas a través de la clase Application.java, que contendrá todos los métodos necesarios para la gestión de las peticiones.

Nombre	Application
Descripción	Clase encargada de responder todas las peticiones realizadas al servicio web a través de las aplicaciones de profesores y de alumnos. Se encargará de la comunicación con el servidor FTP, del tratamiento de los ficheros, y del cálculo de las notas a partir de las evaluaciones realizadas.
Atributos	<ul style="list-style-type: none"> • rutaFTP: Atributo de tipo String que contendrá la dirección del servidor FTP.

	<ul style="list-style-type: none"> • userFTP: nombre de usuario utilizado para acceder al servidor FTP. • passFTP: contraseña del usuario utilizado para acceder al servidor FTP. • rutaProyecto: ruta utilizada dentro del servidor FTP para almacenar toda la información referente a al proyecto. • porcentajeValido: porcentaje utilizado para determinar si la nota válida será la dada por los alumnos o la dada por el profesor. • ftpGen: variable de tipo FTPClient utilizada para establecer y mantener la conexión con el servidor FTP.
Métodos	<ul style="list-style-type: none"> • conectar(): método para conectar con el servidor FTP. • serverIsOn(): método que determinará si es posible establecer conexión con el servidor web. • listarCarpetas(String directorio): devuelve un listado de todas las carpetas existentes en un directorio dado. • listarFicheros(String directorio): devuelve un listado de todos los ficheros existentes en un directorio dado. • descargarFichero(String ruta, String fichero): método que descarga un fichero existente en el servidor FTP. • subirFichero(String ruta, String fichero): método encargado de subir un fichero al servidor FTP. • tipoCuenta(String usuario): método que determinará si la cuenta utilizada es de profesor o de alumno. • listarAsignaturas(String usuario): devuelve un listado de todas las asignaturas disponibles para un usuario. • listarEvaluaciones(String asignatura): método que devolverá las evaluaciones existentes para una asignatura determinada. • evaluaciónRealizada(String asignatura, String evaluación, String usuario): método que determinará si la evaluación

seleccionada ha sido realizada previamente por el usuario.

- **evaluaciónCerrada(String asignatura, String evaluación):** método que determinará si la evaluación seleccionada se encuentra abierta o cerrada.
- **descargarCriterios(String asignatura, String evaluación):** método que descargará del servidor FTP el listado de criterios que contiene una evaluación determinada.
- **enviarEvaluación(String asignatura, String evaluación, JSONObject respuesta, String usuario):** método que enviará al servidor FTP la evaluación realizada por el profesor en formato JSON.
- **actualizarNotaAlumnos(String asignatura, String evaluacion, String grupo, String usuario, String fichero):** método que actualizará la nota dada por el grupo de alumnos en una evaluación dada.
- **actualizarNotaProfesor(String asignatura, String evaluación, String grupo, String usuario, String fichero):** método que actualizará el fichero que contiene la nota dada por el profesor a un grupo de alumnos.
- **descargarEvaluación(String asignatura, String evaluación, String grupo, String usuario):** método que devolverá en formato JSON la evaluación realizada anteriormente por el usuario para la evaluación y el grupo dados.
- **cerrarEvaluacion(String asignatura, String evaluacion):** método que cerrará la evaluación recibida como parámetro.
- **listarGrupos(String asignatura, String evaluación):** método que devolverá un listado de todos los grupos existentes para una evaluación dada.

	<ul style="list-style-type: none"> • listarAlumnos(String asignatura): método que devolverá un listado de todos los alumnos de una asignatura. • obtenerGrupo(String asignatura, String evaluación, String alumno): método que determinará el grupo al que pertenece un alumno dado. • miembrosGrupo(String asignatura, String evaluación, String alumno): método que devuelve los integrantes de un grupo a partir de uno de sus integrantes. Será utilizado en la aplicación de alumno. • miembrosGrupoID(String asignatura, String evaluación, String id): método que devuelve los integrantes de un grupo a partir del identificador del grupo. Será utilizado en la aplicación de profesor. • listarNotas(String asignatura): método que devolverá las evaluaciones que ya tienen la nota disponible. • descargarNotas(String asignatura, String evaluación, String grupo): método que devolverá en formato JSON la nota obtenida por un grupo de alumnos.
--	---

Ilustración 33. Clase Application.

4.4 Diagramas de secuencia

En esta sección se mostrarán los distintos diagramas de secuencia que contendrán las distintas interacciones posibles entre el usuario y la aplicación. Para su realización se utilizará el programa StarUML.

Un diagrama de secuencia muestra la interacción de un conjunto de objetos en una aplicación a través del tiempo y se modela para cada caso de uso. Contiene detalles de implementación del escenario, incluyendo los objetos y clases que se usan para implementar el escenario y mensajes intercambiados entre los objetos. ^[22]

A continuación se mostrarán los diagramas de secuencia para cada aplicación (profesor y alumno) comenzando por la aplicación del profesor.

4.4.1 Diagramas de secuencia. Aplicación profesor.

4.4.1.1 Diagrama de secuencia DS-CU-01-P

Este primer diagrama de secuencia hará referencia al caso de uso uno. En él se mostrarán los pasos que deben seguirse para poder loguearse en la aplicación.

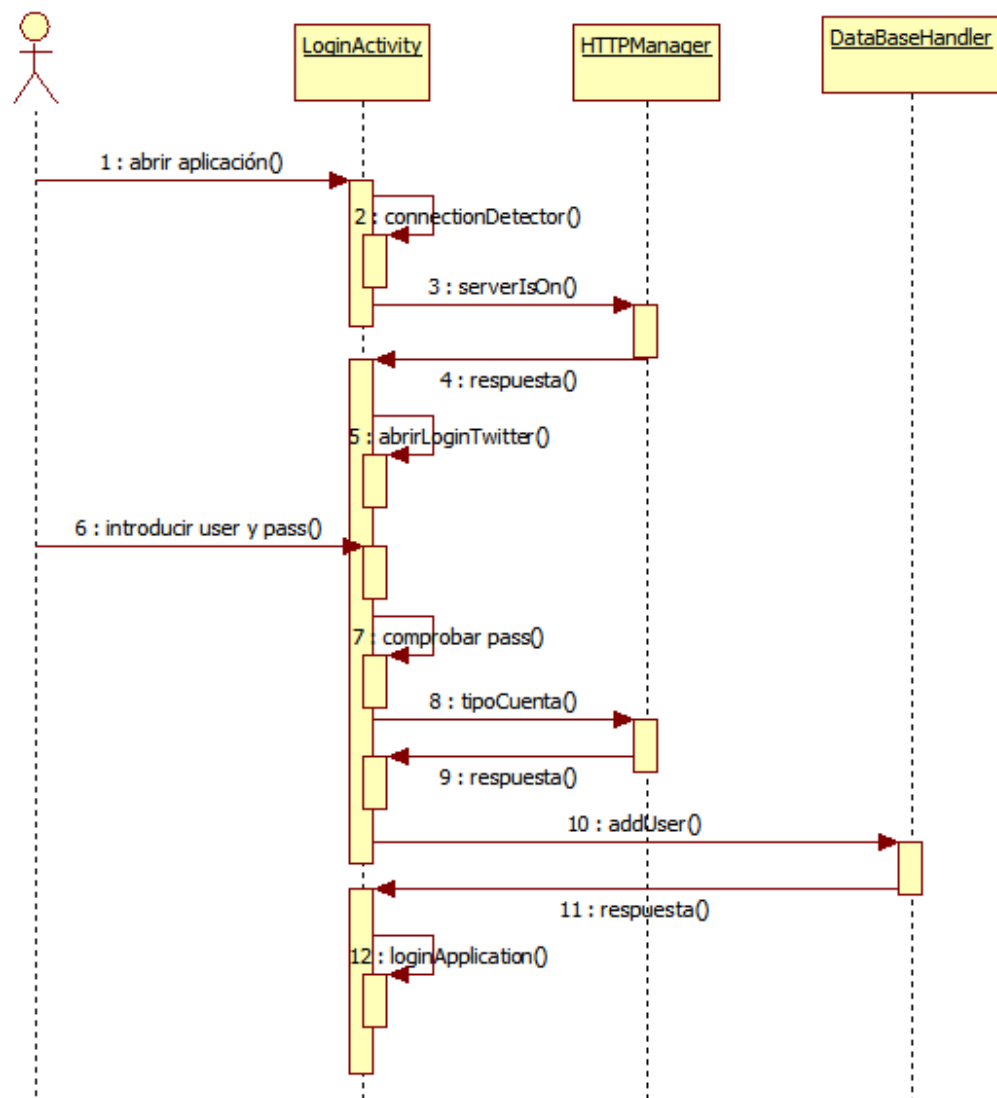


Ilustración 34. DS-CU-01-P

En este diagrama de secuencia en primer lugar se comprueba si existe conexión a internet y si se puede conectarse con el servidor web. Tras esto se pide al usuario que introduzca su usuario y contraseña y se comprueba que los datos son correctos. Si son correctos se comprueba si el usuario es un profesor mediante el servicio web y se añade el usuario a la base de datos. Finalmente se realiza el login en la aplicación.

4.4.1.2 Diagrama de secuencia DSP-CU-03-P

Este diagrama de secuencia hará referencia al caso de uso número 3. En él se mostrarán los pasos que deben seguirse para poder ver el listado de evaluaciones que están disponibles para la asignatura. Este diagrama de secuencia hace referencia a los pasos que deben seguirse cuando el usuario ya está logueado en la aplicación.

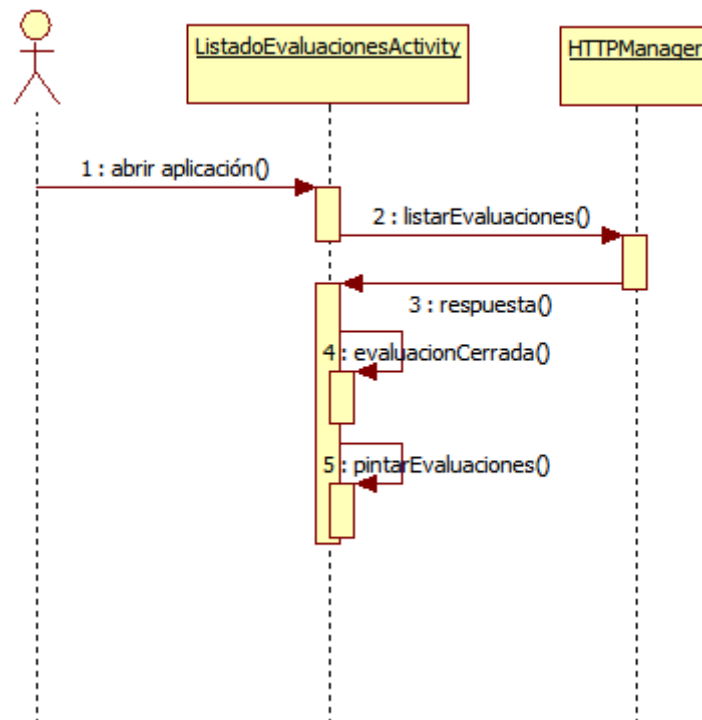


Ilustración 35. DS-CU-03-P

En este diagrama de secuencia en primer lugar el profesor abre la aplicación. En caso de que el usuario no se haya logueado anteriormente deberá realizar los pasos indicados en el diagrama de secuencia DS-CU-01-P, pero para este diagrama de secuencia hemos supuesto que el profesor ya se había logueado anteriormente.

Tras esto se pide a la clase `HTTPManager` que realice una petición al servicio web para que devuelva todas las evaluaciones disponibles para la asignatura impartida por la cuenta con la que se ha realizado el login. Para cada evaluación devuelta, se comprueba mediante el servicio web si se encuentra abierta o cerrada.

Finalmente se pintan todas las evaluaciones para que el usuario pueda visualizarlas.

4.4.1.3 Diagrama de secuencia DS-CU-04-P

Este diagrama de secuencia hará referencia al caso de uso número cuatro. En él se mostrarán los pasos que deben seguirse para poder ver un listado de todos los grupos existentes para una evaluación determinada.

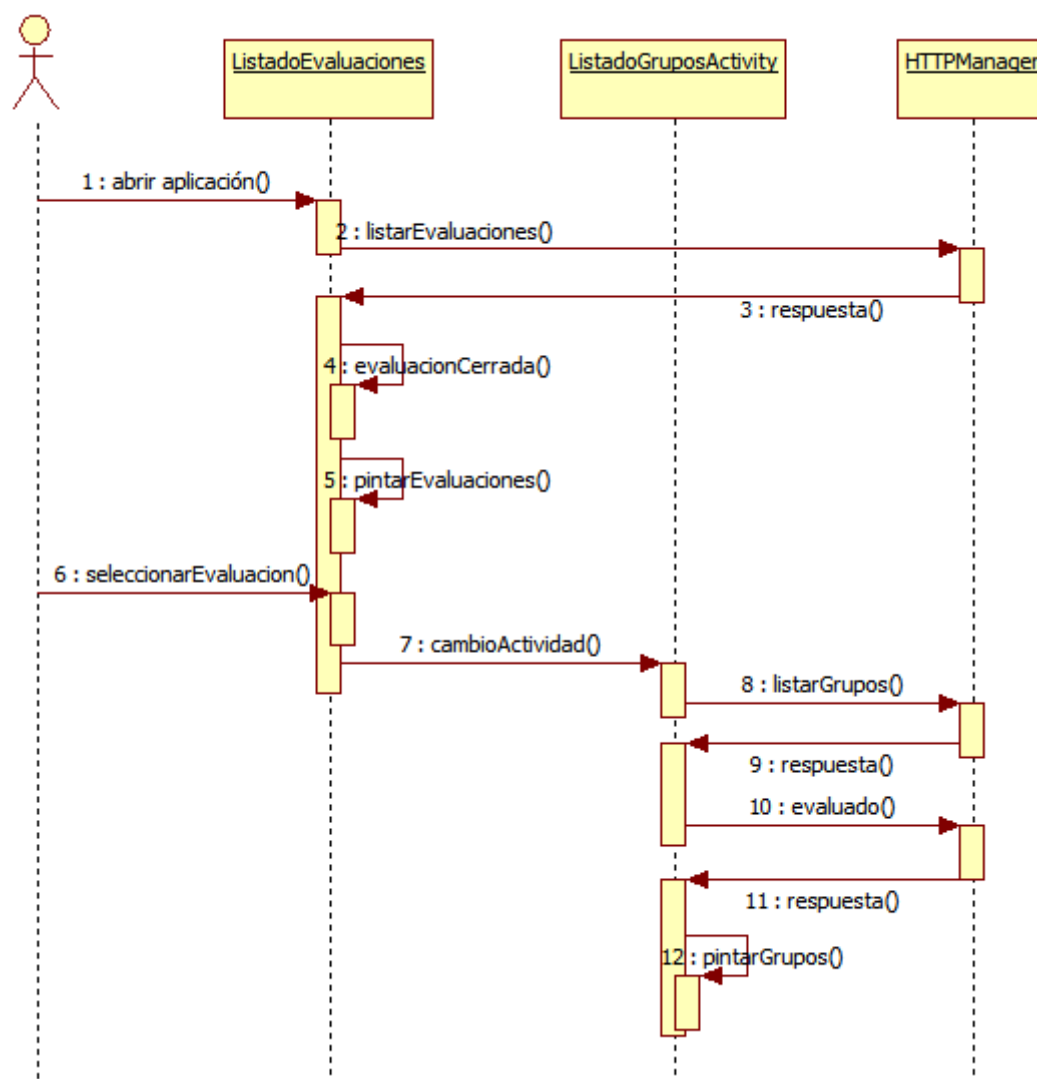


Ilustración 36. DS-CU-04-P

Tras seguir los pasos indicados en el diagrama DS-CU-03-P el profesor seleccionará una evaluación y se producirá un cambio de la actividad en ejecución. La actividad ListadoGruposActivity realizará una petición de los grupos de la evaluación a través de la clase HTTPManager y comprobará qué grupos han sido evaluados anteriormente.

Finalmente se pintarán los grupos para su visualización en pantalla.

4.4.1.4 Diagrama de secuencia DS-CU-05-P

Este diagrama de secuencia hará referencia al caso de uso número cinco. En él se mostrarán los pasos que debe seguir para poder evaluar a uno de los grupos de su asignatura.

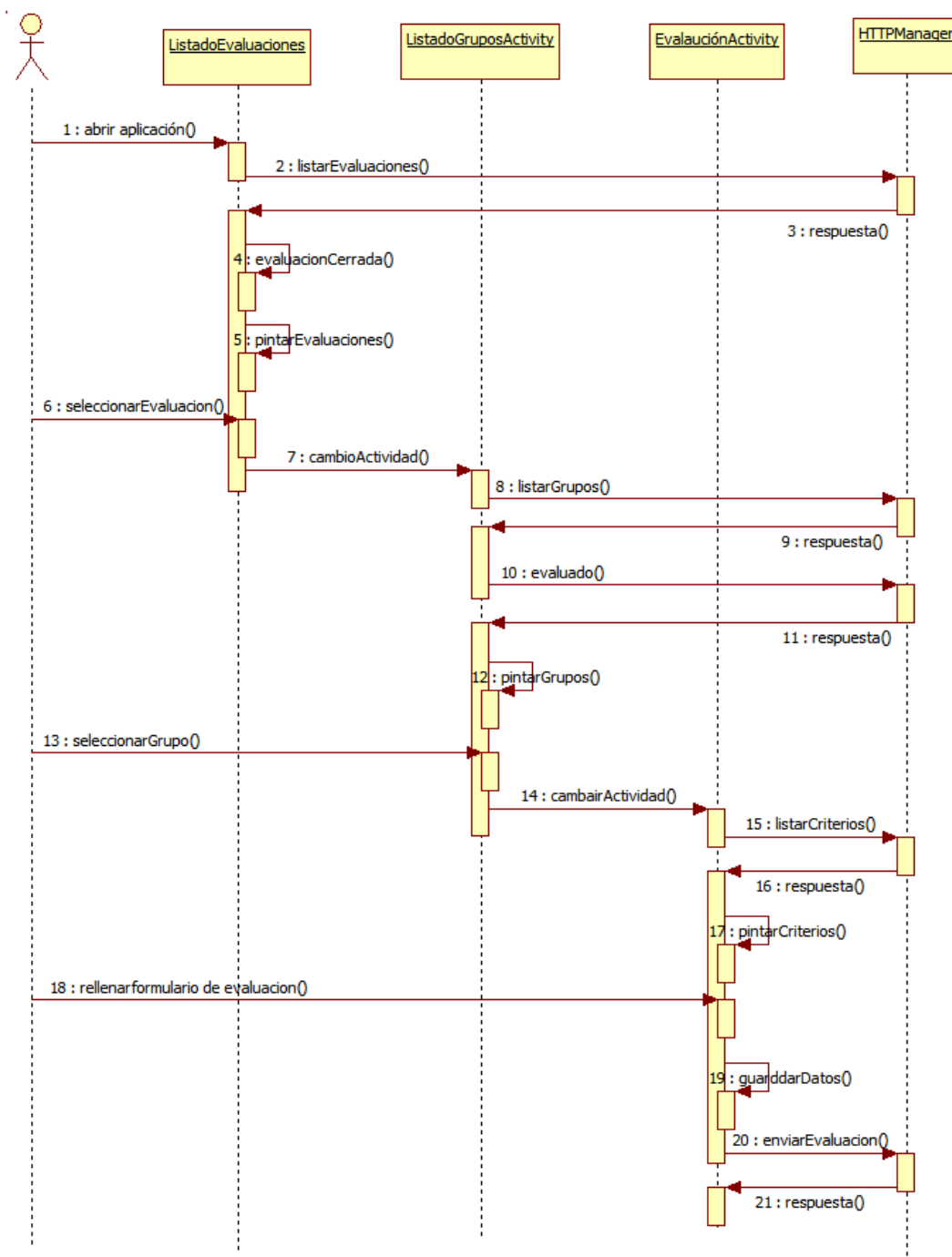


Ilustración 37. DS-CU-05-P

4.4.1.5 Diagrama de secuencia DS-CU-06-P

Este diagrama de secuencia hará referencia al caso de uso número seis. En él se mostrarán los pasos que deben seguirse para poder ver los valores introducidos en una evaluación realizada anteriormente.

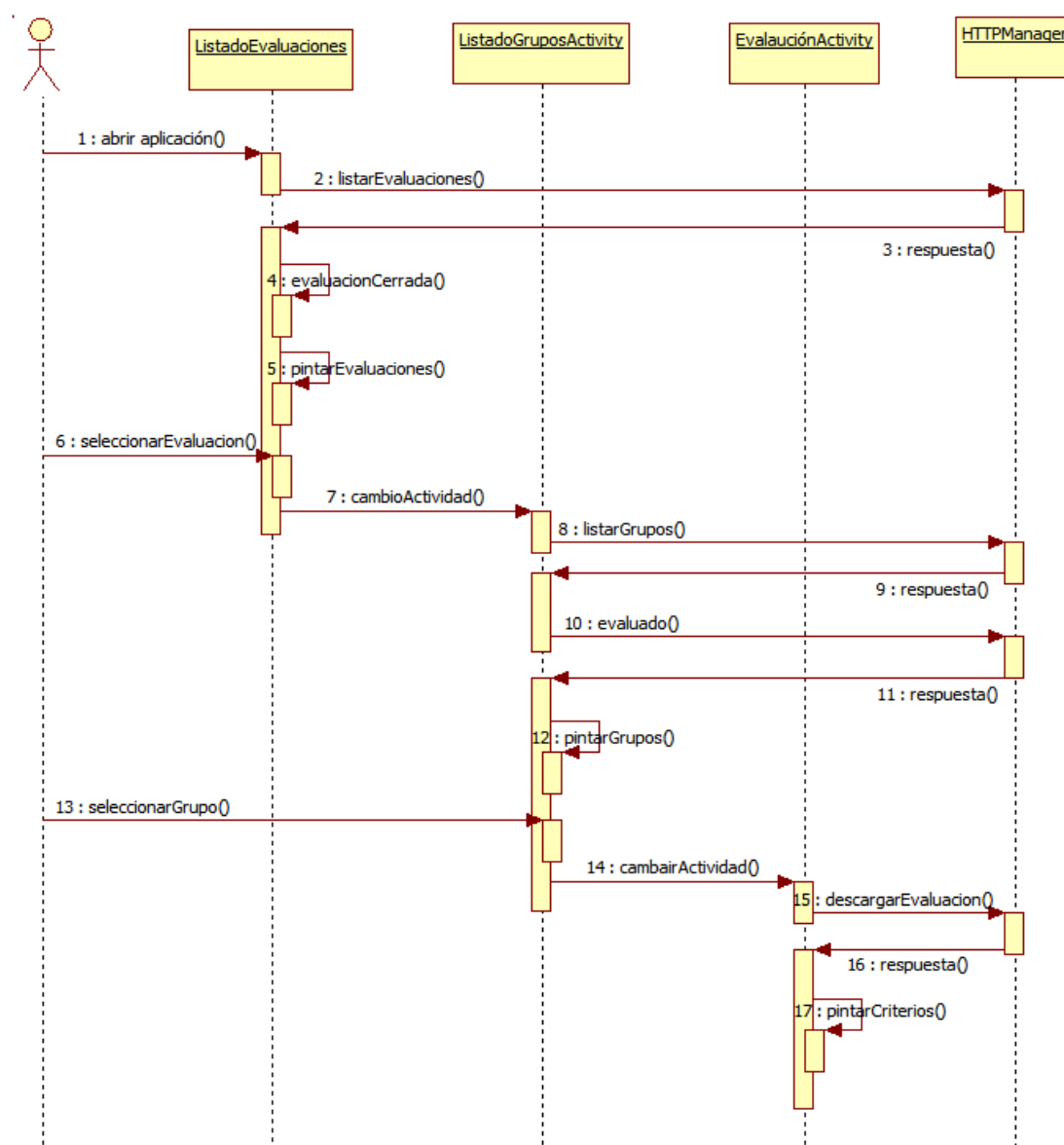


Ilustración 38. DS-CU-06-P

Tras seguir los pasos indicados en el diagrama DS-CU-04-P el profesor seleccionará el grupo del que desea ver la evaluación. Tras esto, se producirá un cambio de la actividad en ejecución y la actividad EvaluacionActivity descargará la evaluación realizada a través de la clase HTTPManager y la mostrará por pantalla.

4.4.1.6 Diagrama de secuencia DS-CU-07-P

Este diagrama de secuencia hará referencia al caso de uso número siete. En él se mostrarán los pasos que deben seguirse para poder cerrar una evaluación.

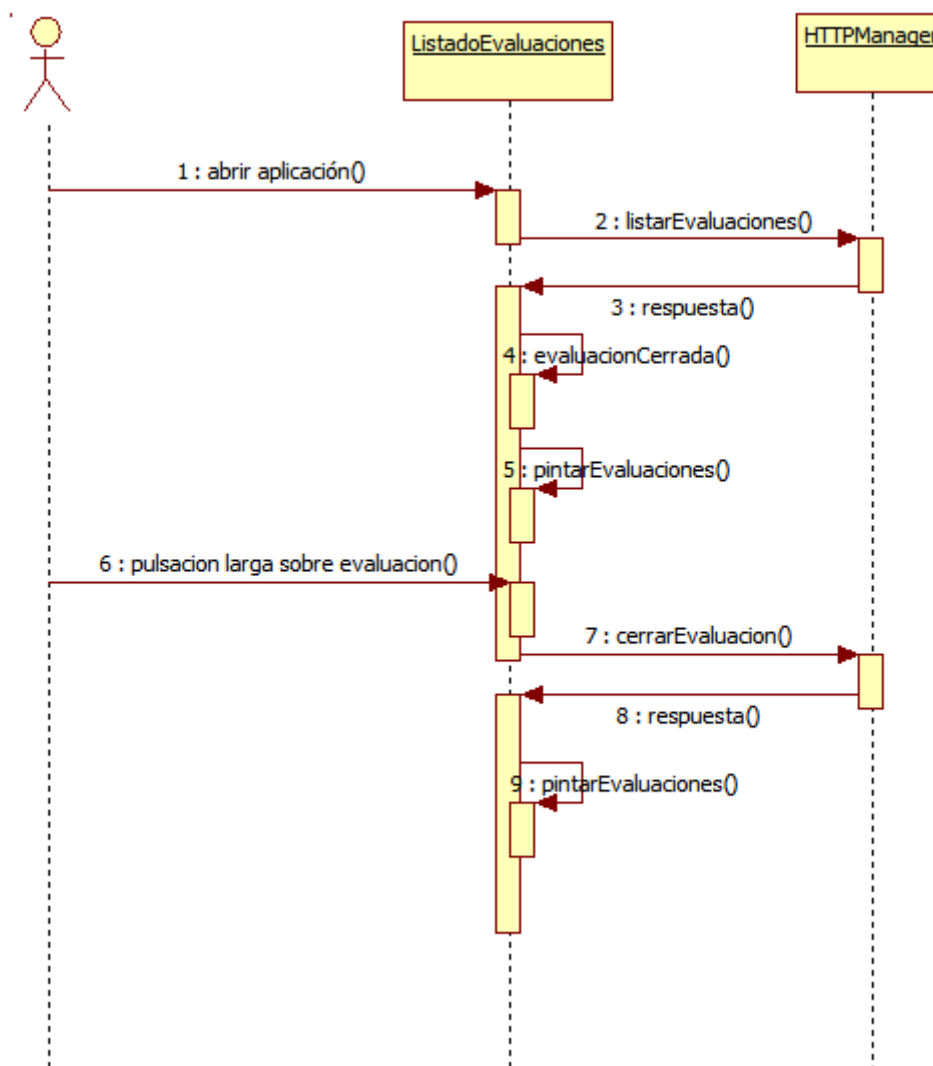


Ilustración 39. DS-CU-07-P

En este diagrama de secuencia, tras seguir los pasos indicados en el diagrama DS-CU-03-P, el profesor realizará una pulsación larga sobre la evaluación que desea cerrar. Tras esto, la clase ListadoEvaluacionesActivity llamará al método cerrarEvaluacion(), que a través de la clase HTTPManager realizará una petición al servicio web para que cierre dicha evaluación.

Finalmente se pintarán de nuevo las evaluaciones disponibles.

4.4.1.7 Diagrama de secuencia DS-CU-08-P

Este diagrama de secuencia hará referencia al caso de uso número ocho. En él se mostrarán los pasos que deben seguirse para poder ver la nota que ha obtenido un grupo determinado para una evaluación dada.

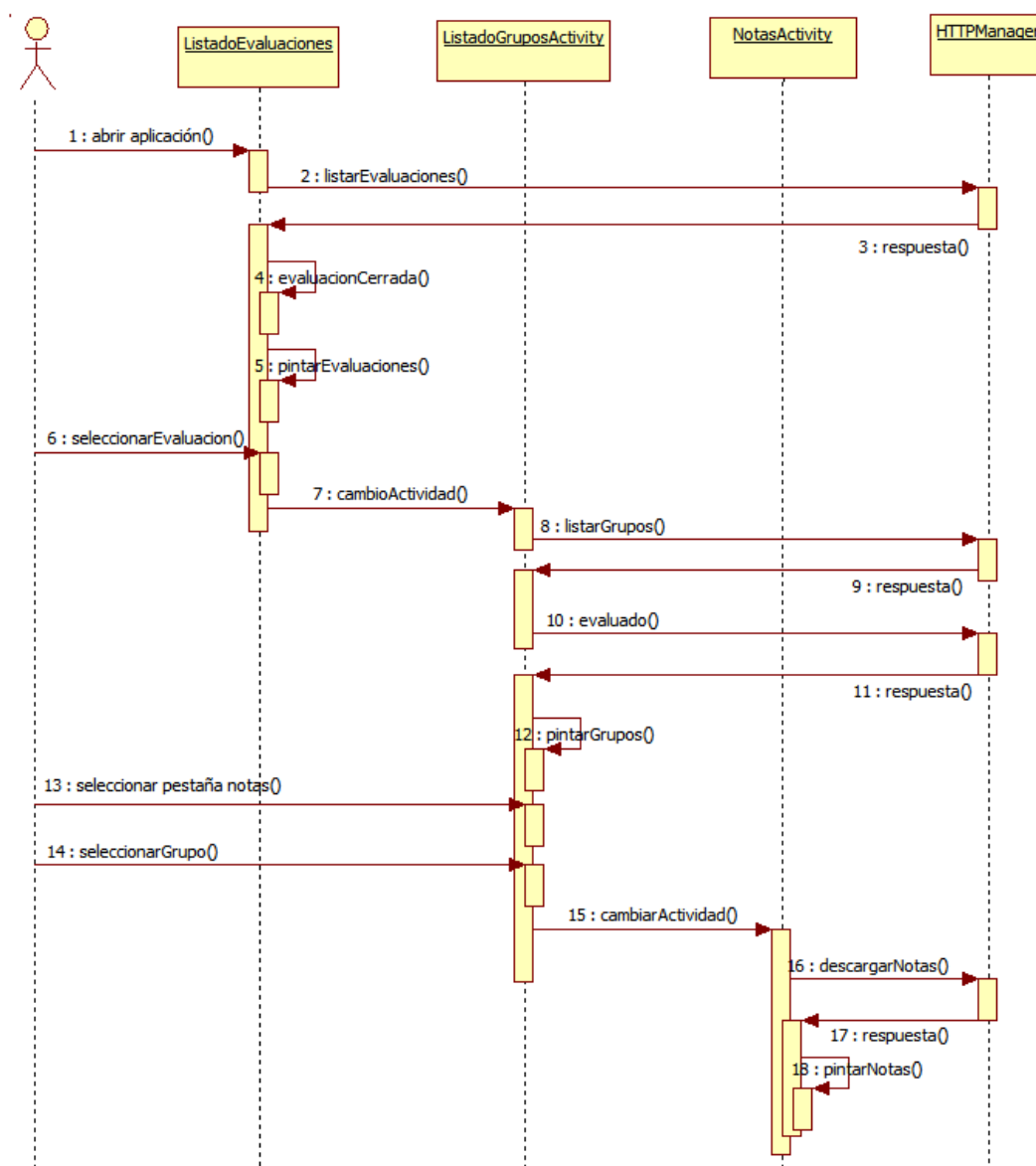


Ilustración 40. DS-CU-08-P

Tras seguir los pasos indicados en el diagrama DS-CU-04-P el profesor seleccionará la pestaña “Notas” e indicará el grupo que desee, produciéndose así un cambio de la actividad en ejecución. Tras esto se descargarán las notas de dicho grupo a través de la clase HTTPManager y se mostrarán las notas por pantalla.

4.4.1.8 Diagrama de secuencia DS-CU-09-P

Este diagrama de secuencia hará referencia al caso de uso número nueve. En él se mostrarán los pasos que deben seguirse para poder ver un listado de todos los alumnos de una asignatura.

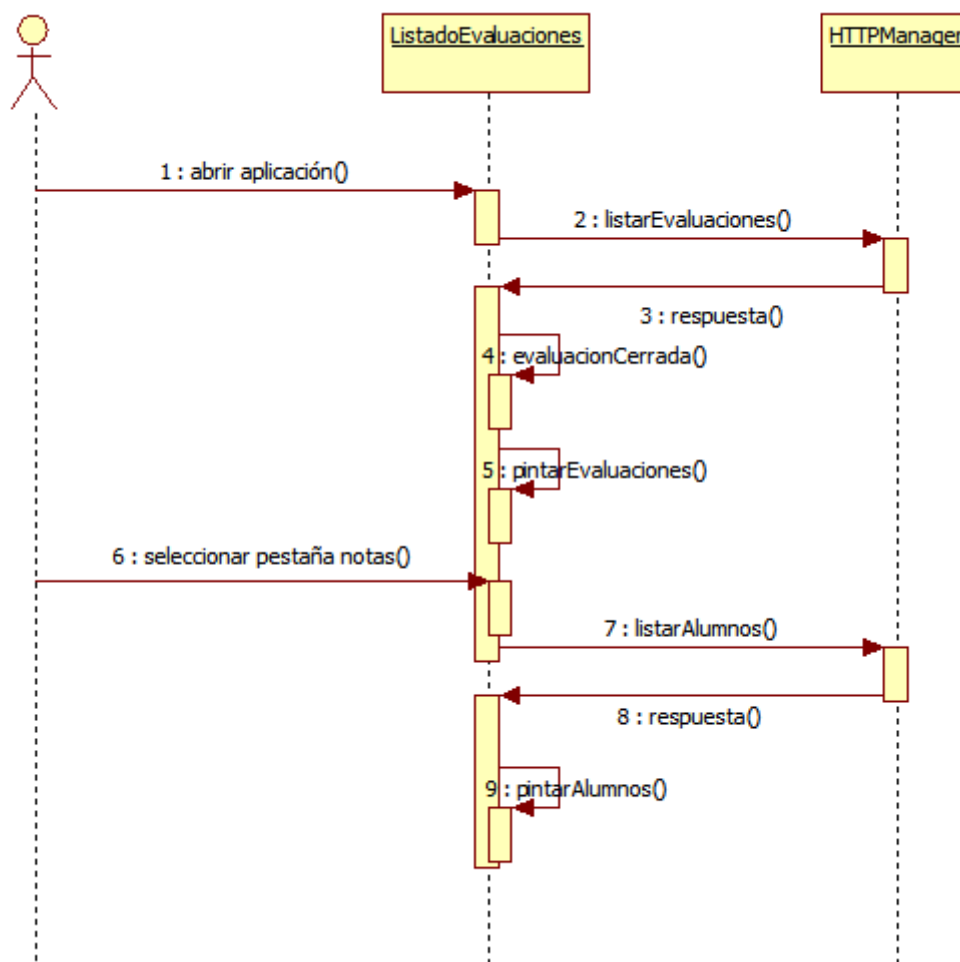


Ilustración 41. DS-CU-09-P

Tras seguir los pasos indicados en el diagrama DS-CU-03-P, el profesor seleccionará la pestaña “Notas”. Tras esto, se realizará una petición del listado de alumnos al servicio web a través de la clase HTTPManager.

Finalmente se llamará al método `pintarAlumnos()` para que el profesor pueda visualizar en la pantalla el listado de alumnos de la asignatura.

4.4.1.9 Diagrama de secuencia DS-CU-10-P

Este diagrama de secuencia hará referencia al caso de uso número diez. En él se mostrarán los pasos que deben seguirse para poder ver todas las notas que ha obtenido un alumno en una asignatura.

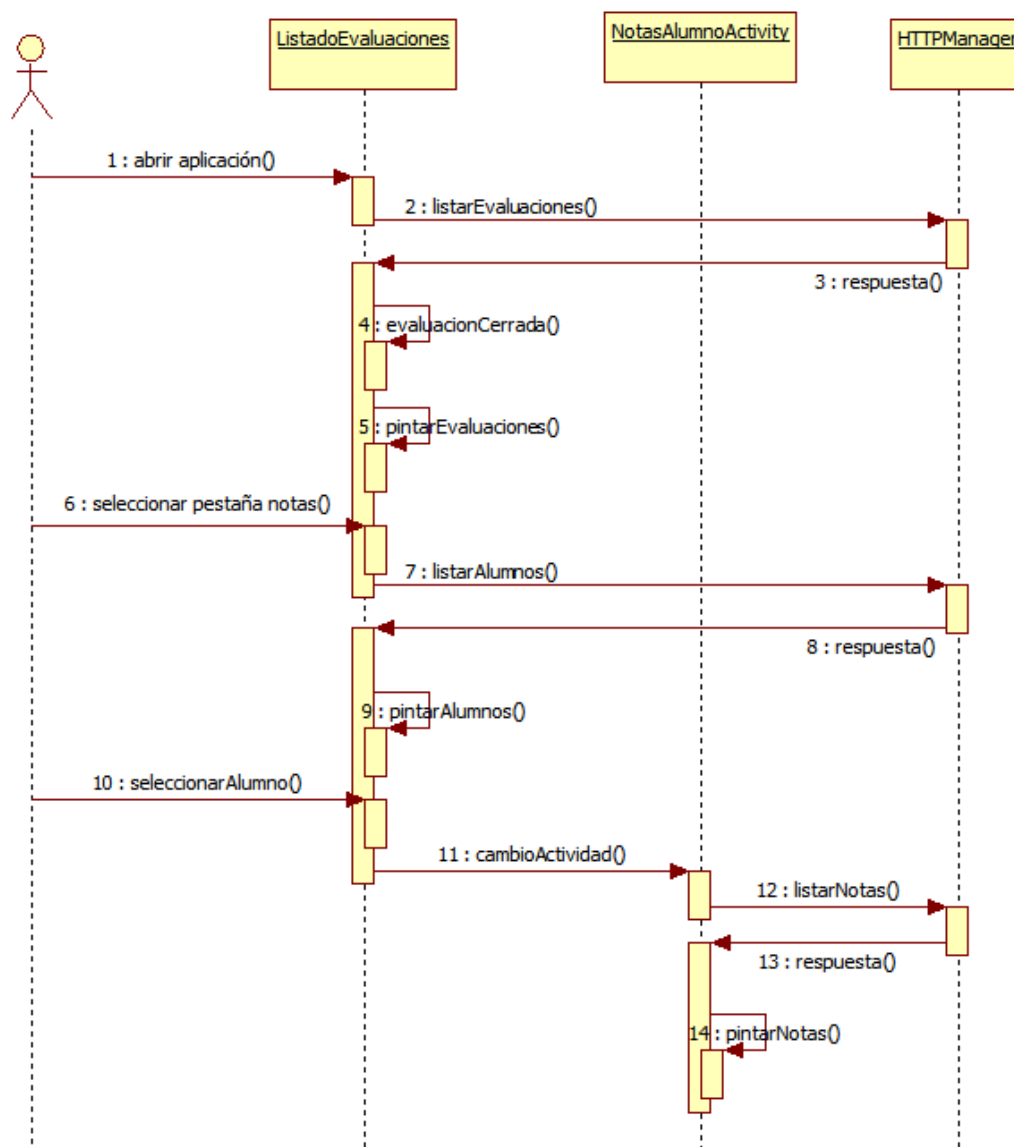


Ilustración 42. DS-CU-10-P

Tras seguir los pasos indicados en el diagrama DS-CU-09-P el profesor seleccionará un alumno. Tras esto se producirá un cambio en la actividad en ejecución y se llamará al método listarNotas() a través de la clase HTTPManager.

Finalmente se pintarán las notas en pantalla.

4.4.1.10 Diagrama de secuencia DS-CU-11-P

Este diagrama de secuencia hará referencia al caso de uso número once. En él se mostrarán los pasos que deben seguirse para cerrar la sesión en la aplicación.

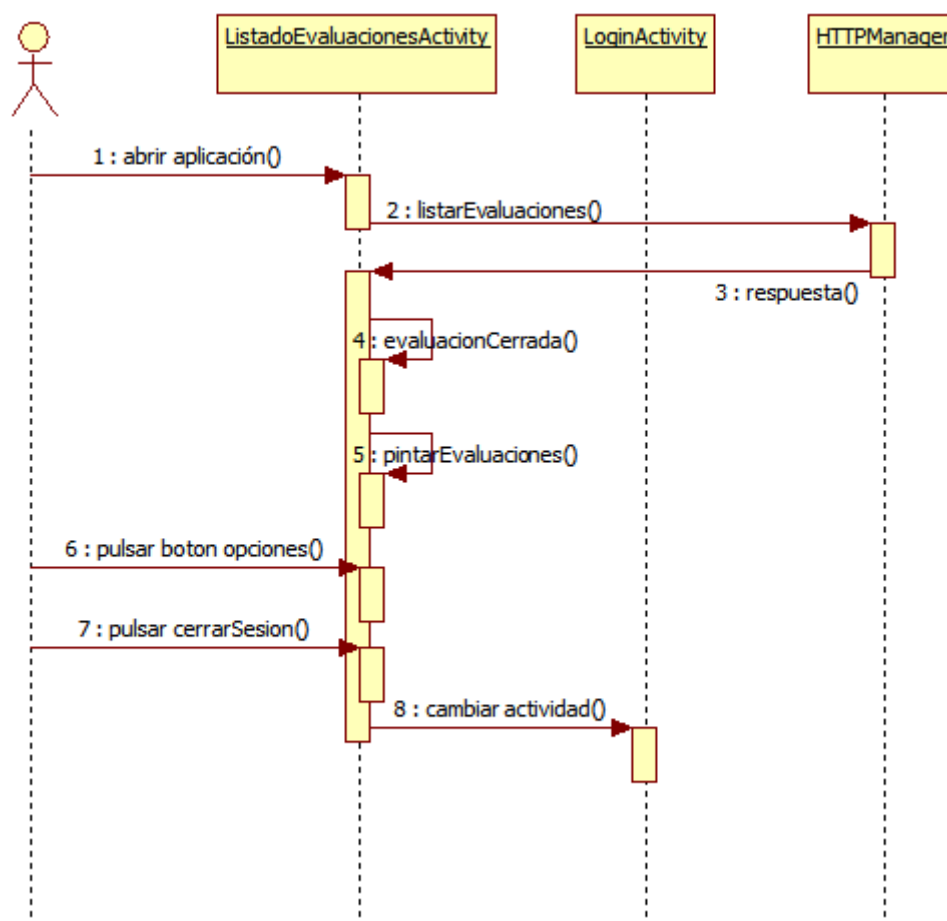


Ilustración 43. DS-CU-11-A

En este diagrama de secuencia, tras completar los pasos indicados en el diagrama DS-CU-03-P el profesor pulsará el botón de opciones del terminal y aparecerá el menú de opciones. Tras esto el profesor seleccionará la opción cerrar sesión.

Finalmente la aplicación volverá a la pantalla de login.

4.4.2 Diagramas de secuencia. Aplicación alumno.

4.4.2.1 Diagrama de secuencia DSA-CU-01-A

Este primer diagrama de secuencia hará referencia al caso de uso uno. En él se mostrarán los pasos que deben seguirse para poder loguearse en la aplicación.

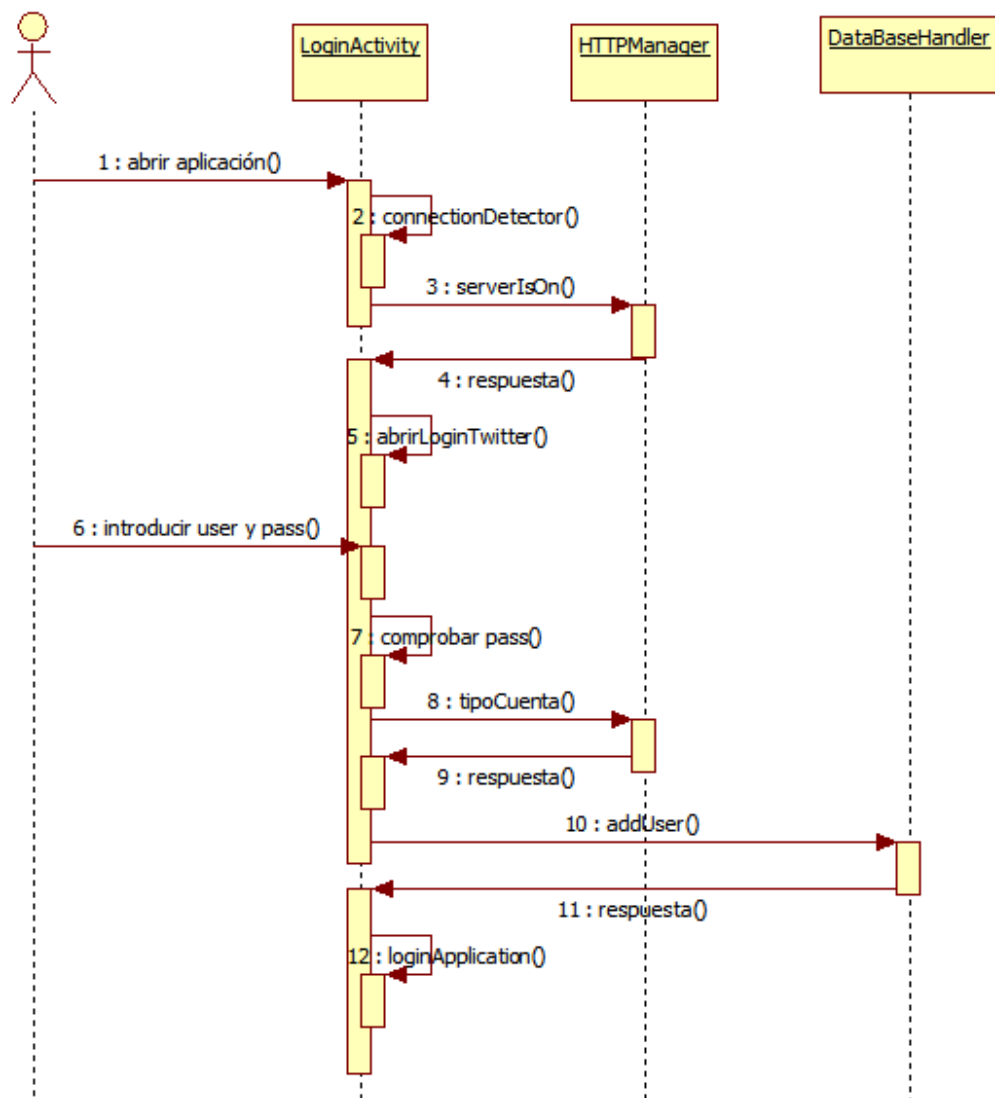


Ilustración 44. DS-CU-01-A

En este diagrama de secuencia en primer lugar se comprueba si existe conexión a internet y si es posible conectar con el servidor web. Tras esto se pide al usuario que introduzca su usuario y contraseña y se comprueba que los datos son correctos. Si son correctos se comprueba si el usuario es un alumno mediante el servicio web, se añade el usuario a la base de datos y se realiza el login en la aplicación.

4.4.2.2 Diagrama de secuencia DS-CU-02-A

Este diagrama de secuencia hará referencia al caso de uso número dos. En él se mostrarán los pasos que deben seguirse para poder ver el listado de las asignaturas que están disponibles para el alumno.

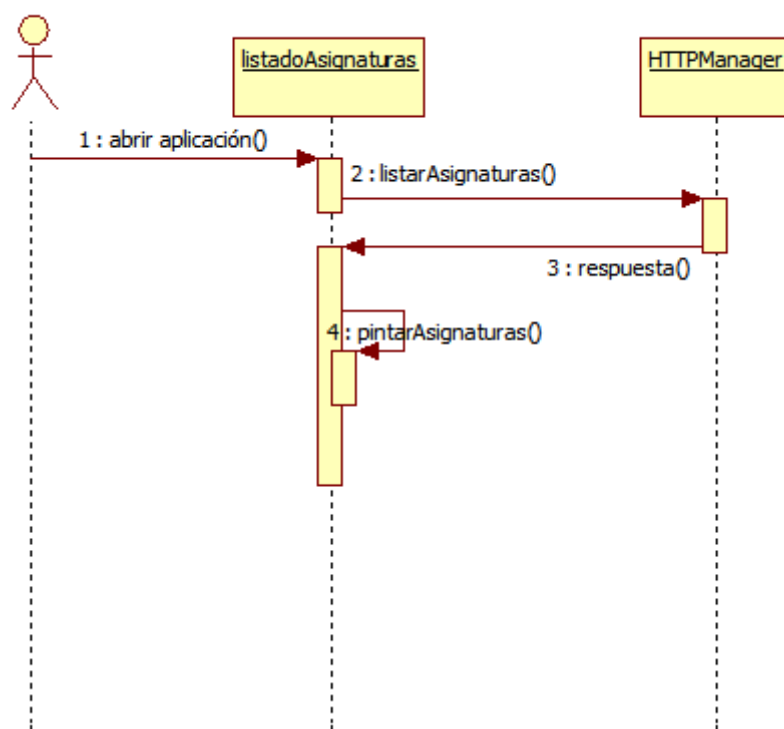


Ilustración 45. DS-CU-02-A

En este diagrama de secuencia en primer lugar el alumno abre la aplicación. En caso de que el usuario no se haya logueado anteriormente deberá realizar los pasos indicados en el diagrama de secuencia DS-CU-01-P. Para este diagrama de secuencia hemos supuesto que el alumno ya se había logueado anteriormente.

Tras esto se pide a la clase `HTTPManager` que realice una petición al servicio web para que devuelva todas las asignaturas disponibles para el alumno logueado en la aplicación.

Finalmente se pintan todas las evaluaciones para que el usuario pueda visualizarlas.

4.4.2.3 Diagrama de secuencia DS-CU-03-A

Este diagrama de secuencia hará referencia al caso de uso número tres. En él se mostrarán los pasos que deben seguirse para poder ver el listado de las evaluaciones que están disponibles para la asignatura seleccionada.

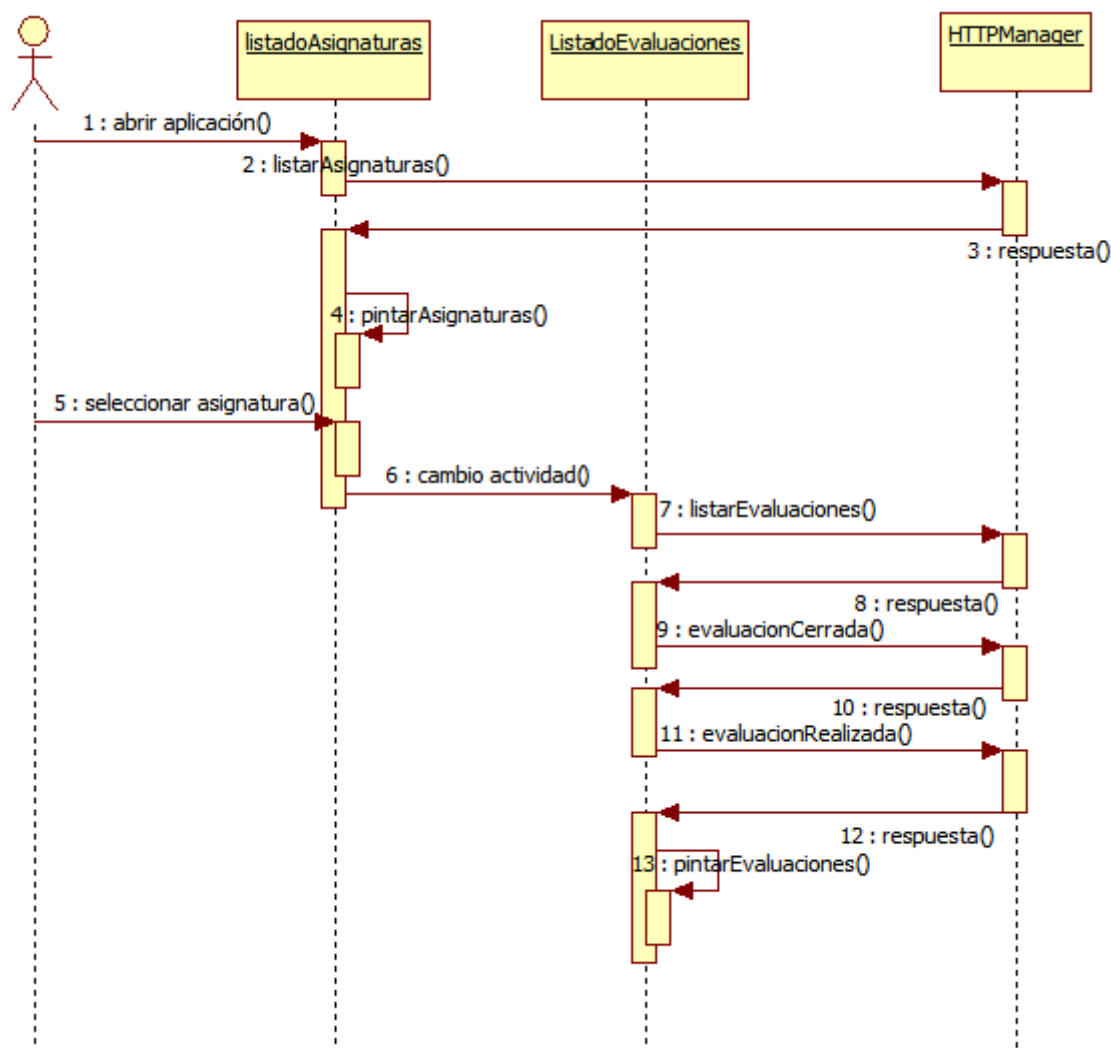


Ilustración 46. DS-CU-03-A

Tras completar los pasos indicados en el diagrama DS-CU-02-A, el alumno seleccionará la asignatura que desee. Tras esto pasaremos a la actividad ListadoEvaluacionesActivity, que pedirá el listado de evaluaciones a través de la clase HTTPManager. Tras esto se comprobará qué evaluaciones están abiertas y cerradas a través de los métodos `evaluacionCerrada()` y `evaluación realizada`.

Finalmente se mostrarán las evaluaciones en pantalla.

4.4.1.4 Diagrama de secuencia DS-CU-05-A

Este diagrama de secuencia hará referencia al caso de uso número cinco. En él se mostrarán los pasos que deben seguirse para poder evaluar al grupo de alumnos al que pertenece el usuario.

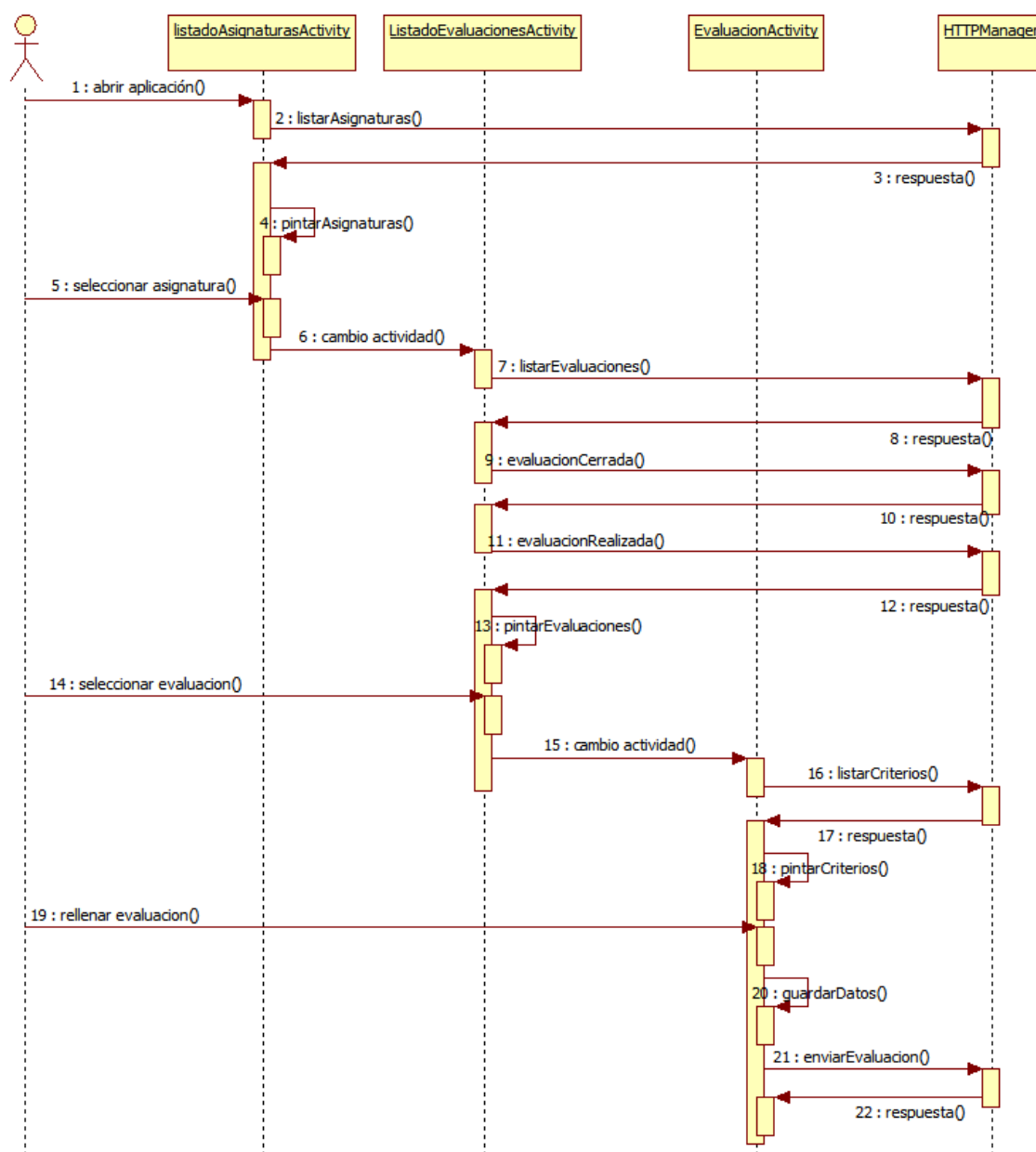


Ilustración 47. DS-CU-05-A

Tras completar los pasos del diagrama DS-CU-03-A el alumno seleccionará una evaluación. Tras esto se solicitarán y pintarán los criterios de la evaluación y el alumno completará el formulario de evaluación. Finalmente la evaluación será enviada.

4.4.1.5 Diagrama de secuencia DS-CU-06-A

Este diagrama de secuencia hará referencia al caso de uso número seis. En él se mostrarán los pasos que deben seguirse para poder ver los valores introducidos en una evaluación realizada anteriormente.

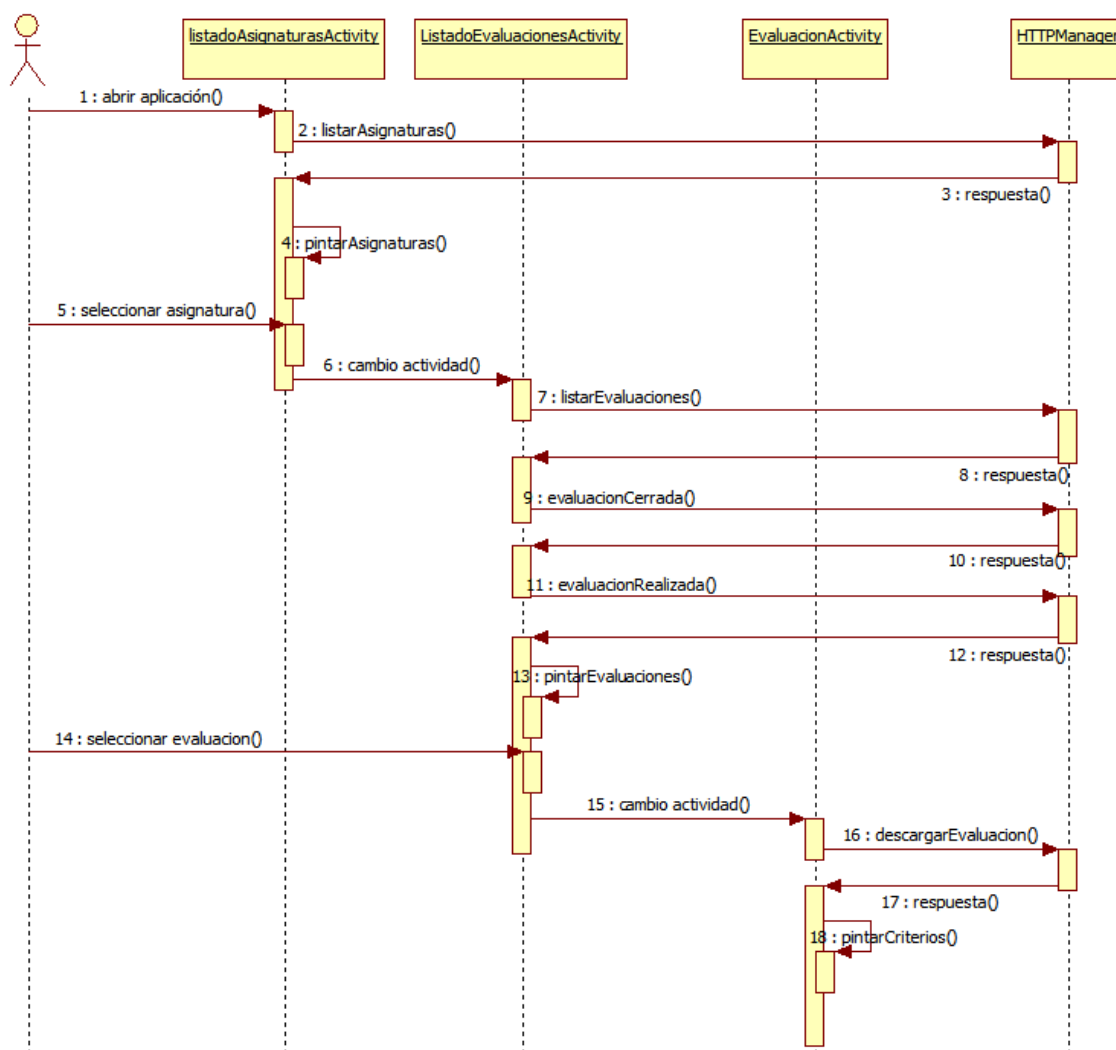


Ilustración 48. DS-CU-08-A

Tras completar los pasos del diagrama DS-CU-03-A el alumno seleccionará una evaluación. Tras esto se cambiará la actividad en ejecución y se descargará la evaluación realizada por el alumno.

Finalmente se pintarán los criterios en la pantalla para que el usuario pueda visualizarlos.

4.4.1.7 Diagrama de secuencia DS-CU-10-A

Este diagrama de secuencia hará referencia al caso de uso número diez. En él se mostrarán los pasos que deben seguirse para poder visualizar la nota que ha obtenido el alumno en una evaluación determinada.

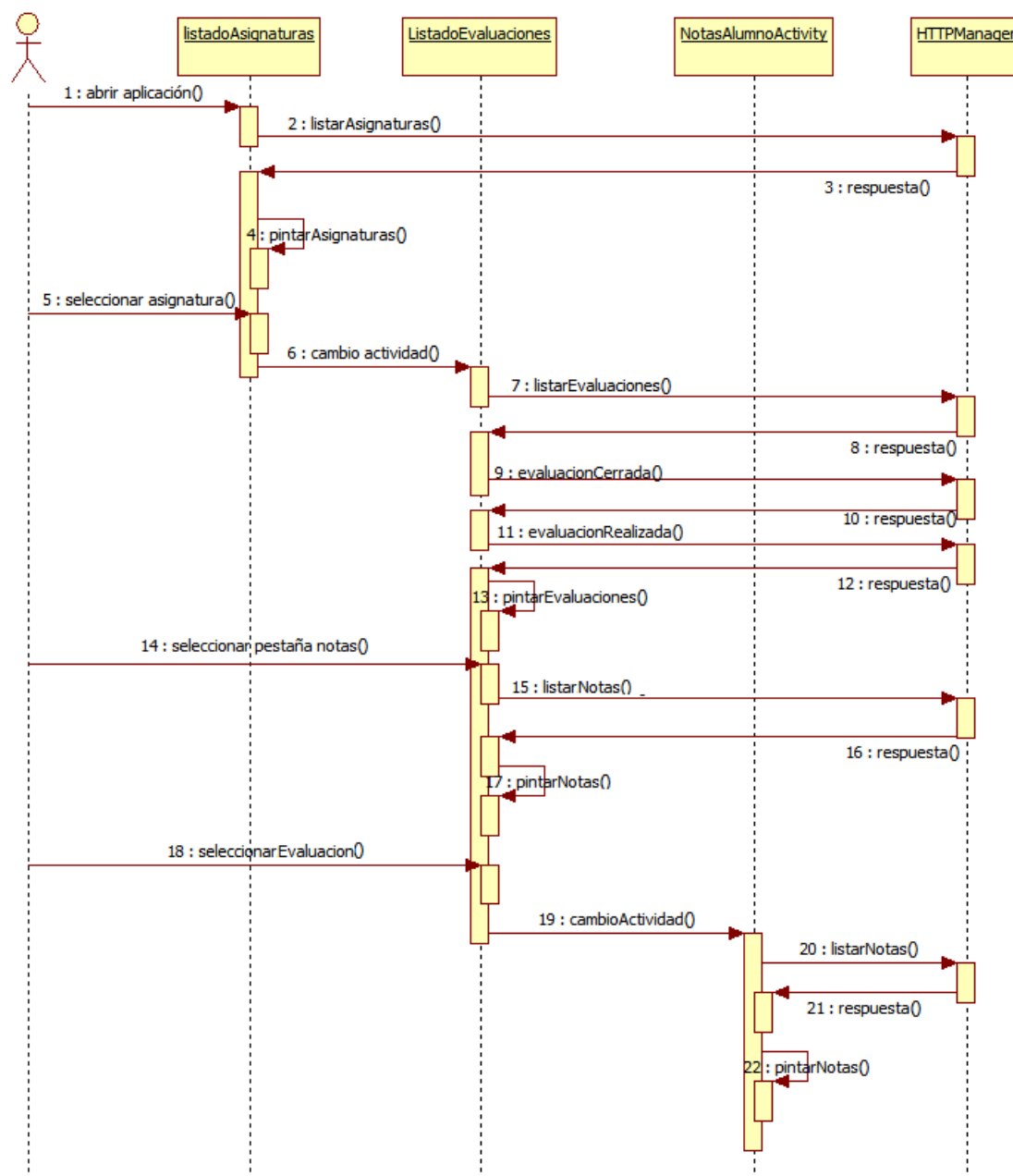


Ilustración 49. DS-CU-10-A

Tras completar los pasos indicados en el diagrama DS-CU-03-A el alumno seleccionará la pestaña “Notas”. Tras esto se listarán las evaluaciones en pantalla y se seleccionará la deseada. Finalmente se mostrarán las notas de dicha evaluación.

4.4.1.8 Diagrama de secuencia DSP-CU-11-A

Este diagrama de secuencia hará referencia al caso de uso número nueve. En él se mostrarán los pasos que deben seguirse para cerrar la sesión en la aplicación.

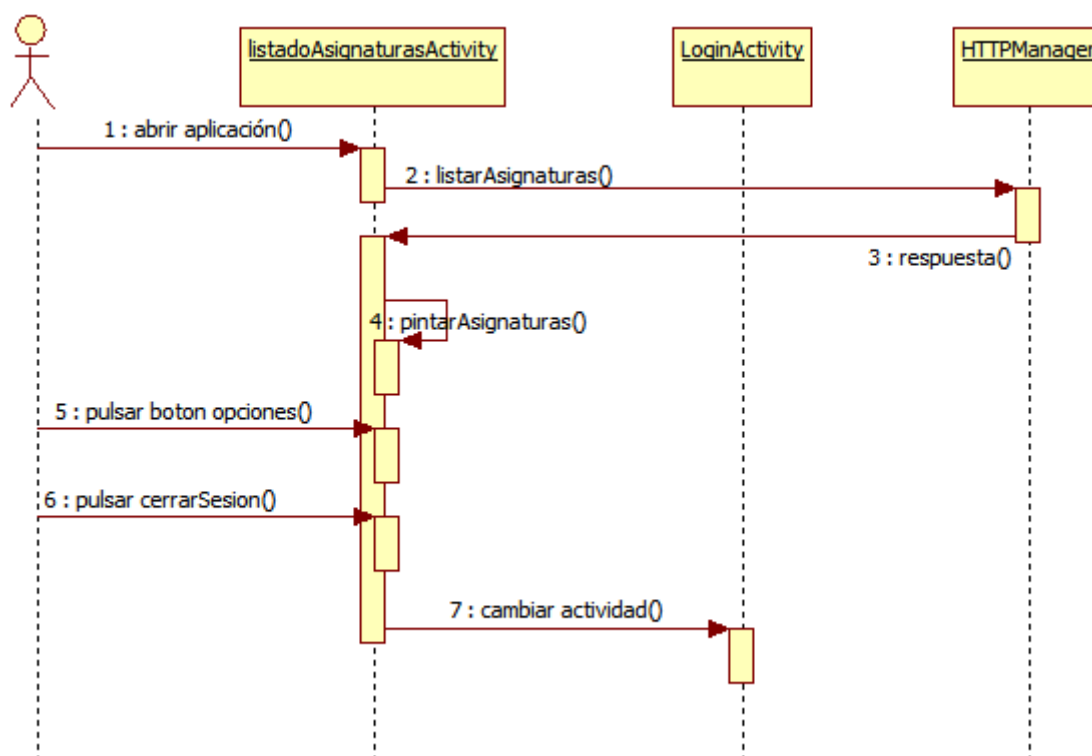


Ilustración 50. DS-CU-11-A

En este diagrama de secuencia, tras completar los pasos indicados en el diagrama DS-CU-02-A el alumno pulsará el botón de opciones del terminal y aparecerá el menú de opciones. Tras esto el alumno seleccionará la opción cerrar sesión. Finalmente la aplicación volverá a la pantalla de login.

4.5 Diseño del servidor FTP

El servidor FTP será utilizado para el almacenamiento de toda la información utilizada en la aplicación.

Como ya hemos dicho anteriormente, el servidor FTP utilizado será un servidor que ya se encuentra en funcionamiento y que será cedido por la universidad.

Contendrá las asignaturas existentes, los alumnos de cada asignatura, las evaluaciones disponibles en cada una de ellas, los criterios que se deben evaluar en cada evaluación, los grupos de alumnos que forman parte de cada evaluación, las evaluaciones enviadas tanto por profesores como por alumnos y las notas obtenidas una vez finalizada la evaluación.

El servidor FTP presentará la siguiente estructura:

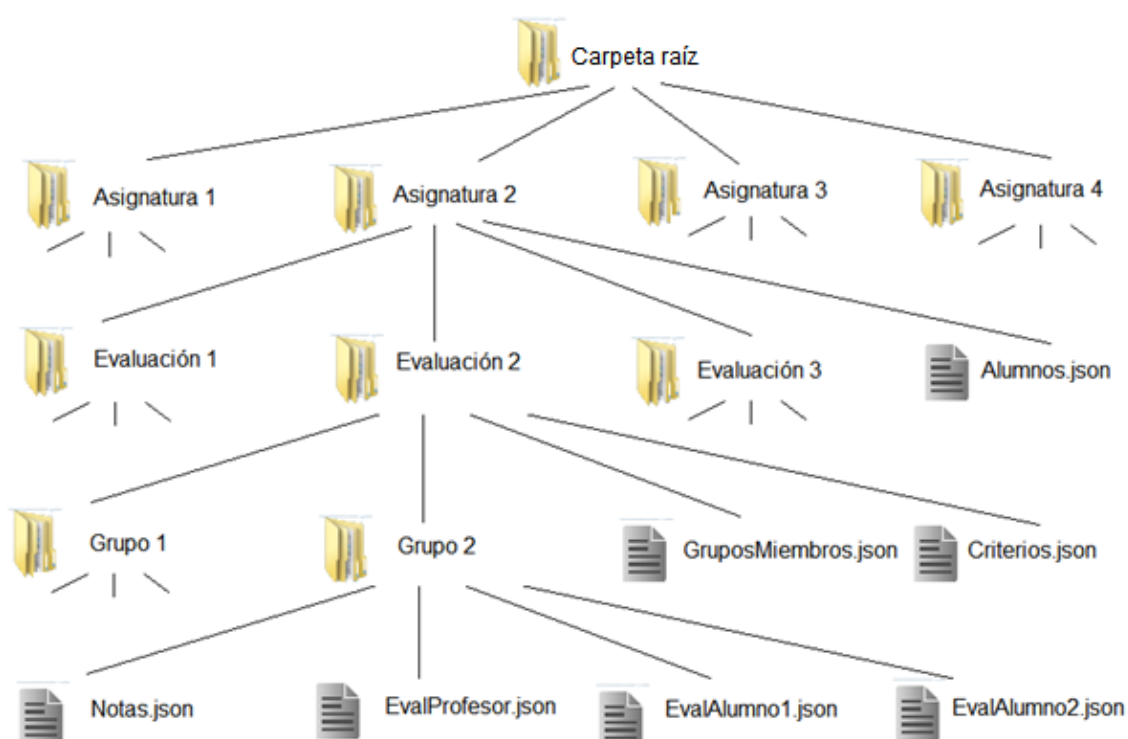


Ilustración 51. Estructura FTP.

Como podemos observar en la imagen, en el nivel más alto se encontrará la carpeta raíz.

Dentro de esta carpeta existirá una carpeta por cada una de las asignaturas disponibles.

Para cada asignatura, existirán distintas evaluaciones, y para cada una de estas evaluaciones deberá existir una nueva carpeta que las represente. Además, en la carpeta de una asignatura deberá existir también un fichero json con el nombre Alumnos.json, que contendrá el listado de todos los alumnos que forman parte de la asignatura.

Cada evaluación deberá contener una nueva carpeta por cada grupo existente en la asignatura (estas carpetas serán generadas automáticamente desde la aplicación). Además, deberán existir dos ficheros json: Criterios.json (que contendrá los distintos criterios y grupos de criterios que se deben tener en cuenta a la hora de realizar una evaluación) y GruposMiembros.json (que contendrá los distintos grupos de alumnos que forman parte de dicha evaluación).

Dentro de la carpeta de cada grupo y a medida que tanto alumnos como profesores vayan realizando sus correspondientes evaluaciones, irán creándose distintos ficheros json. Por cada evaluación realizada por un alumno o un profesor se creará un fichero json que contenga dicha evaluación. Además, tras la realización de la primera evaluación del grupo se creará el fichero Notas.json, que contendrá la nota obtenida por cada uno de los alumnos hasta el momento y que se irá actualizando a medida que se realicen nuevas evaluaciones.

Capítulo V. Pruebas de sistema

A lo largo de este capítulo se expondrán las distintas pruebas realizadas para verificar el correcto funcionamiento del sistema.

Las pruebas serán realizadas tanto en la aplicación de profesores como en la aplicación de alumnos.

Para la realización de pruebas, el servidor FTP ha sido anteriormente provisto de los ficheros necesarios.

Para la exposición de las pruebas se utilizará el siguiente modelo:

Identificador	
Aplicación	
Descripción	
Precondiciones	
Pasos	
Salida esperada	

Tabla 70. Modelo pruebas.

Identificador: Etiqueta que identifica de manera única e inequívoca una prueba. Presentará el formato PR-XX, donde XX representa el número de prueba realizada.

Aplicación: Aplicación donde se ha realizado la prueba. Podrá tomar el valor Profesor o Alumno.

Descripción: Descripción de la prueba que se quiere realizar.

Precondiciones: Conjunto de condiciones que deberán darse antes de realizar la prueba.

Pasos: Conjunto de acciones que hay que realizar para llevar a cabo la prueba.

Salida esperada: campo que indicará que debe suceder o que debe mostrar la aplicación una vez seguidos los pasos del campo anterior.

A continuación se muestran las pruebas realizadas ordenadas por casos de uso:

5.1 Pruebas de sistema: CU-01

A continuación se mostrarán las pruebas relacionadas con el caso de uso CU-01, que tendrán que ver con el login en la aplicación tanto para profesores como para alumnos.

Identificador	PR-01
Aplicación	Profesor
Descripción	Al abrir la aplicación por primera vez deberá mostrarse la pantalla de login con un botón que permitirá loguearse en twitter.
Precondiciones	No haberse logueado previamente en la aplicación.
Pasos	<ul style="list-style-type: none">• Abrir la aplicación por primera vez o sin haberse logueado previamente.
Salida esperada	Pantalla de login que mostrará únicamente un botón para poder loguearse en twitter.

Tabla 71. PR-01

Identificador	PR-02
Aplicación	Profesor
Descripción	Al pulsar el botón para loguearse con un nuevo usuario o loguearse por primera vez, la aplicación nos redireccionará a la página de autenticación de twitter, donde podremos introducir el usuario y la contraseña de la cuenta que queremos utilizar.
Precondiciones	-
Pasos	<ul style="list-style-type: none">• Abrir la aplicación.• Pulsar el botón de login en twitter.
Salida esperada	Se mostrará la pantalla de autenticación de twitter que permitirá introducir el usuario y la contraseña para poder utilizar la aplicación.

Tabla 72. PR-02

Identificador	PR-03
Aplicación	Profesor
Descripción	Al introducir un usuario y contraseña correctas y que pertenezcan a una cuenta de profesor, la aplicación mostrará la pantalla con las distintas evaluaciones.
Precondiciones	Añadir en el servidor FTP una asignatura que se corresponda con la cuenta con la que el profesor ha realizado el login en la aplicación.
Pasos	<ul style="list-style-type: none"> • Abrir la aplicación. • Pulsar el botón de login. • Introducir un usuario correspondiente a una cuenta de profesor.
Salida esperada	Se mostrará la actividad que contendrá las distintas evaluaciones.

Tabla 73. PR-03

Identificador	PR-04
Aplicación	Profesor
Descripción	Al introducir un usuario que no es un profesor deberá mostrarse un aviso que indique que esa cuenta es una cuenta de alumno y que el login debe realizarse con una cuenta de profesor.
Precondiciones	En el servidor FTP no deberá existir ninguna asignatura que se corresponda con la asignatura utilizada para realizar el login.
Pasos	<ul style="list-style-type: none"> • Abrir la aplicación. • Pulsar el botón de login. • Introducir un usuario que no sea un profesor y su correspondiente contraseña.
Salida esperada	Pantalla de login en la que aparecerá un aviso que indicará “Esta cuenta es una cuenta de alumno. Para loguearse mediante esta cuenta descargue la aplicación de alumno”.

Tabla 74. PR-04

Identificador	PR-05
Aplicación	Profesor
Descripción	Al abrir la aplicación, cuando ya nos hemos logueado previamente con un usuario y accedemos a la pantalla de login, además del botón de login en twitter aparecerá un desplegable con todas las cuentas introducidas anteriormente y un botón para loguearse con un nuevo usuario.
Precondiciones	Haberse logueado con al menos un usuario y después haber cerrado sesión.
Pasos	<ul style="list-style-type: none"> • Abrir la aplicación.
Salida esperada	Pantalla de login con un listado de las cuentas con las que nos hemos logueado anteriormente, un botón para loguearse con uno de ellos y otro botón que permitirá loguearse con un nuevo usuario.

Tabla 75. PR-05

Identificador	PR-06
Aplicación	Profesor
Descripción	Al acceder a la aplicación cuando ya estamos logueados con una cuenta de profesor se mostrará directamente la pantalla con las distintas evaluaciones en lugar de la pantalla de login.
Precondiciones	Estar logueado en la aplicación con una cuenta de profesor.
Pasos	<ul style="list-style-type: none"> • Abrir la aplicación.
Salida esperada	Se mostrará la interfaz que contendrá un listado con todas las evaluaciones disponibles para la asignatura impartida por el profesor que se ha logueado en la aplicación.

Tabla 76. PR-06

Identificador	PR-07
Aplicación	Alumno
Descripción	Al abrir la aplicación por primera vez, antes de haberse logueado con ningún usuario, la aplicación deberá mostrar la pantalla de login con un botón que permitirá Loguearse en twitter.
Precondiciones	No haberse logueado previamente en la aplicación
Pasos	<ul style="list-style-type: none"> • Abrir la aplicación por primera vez o sin haberse logueado previamente.
Salida esperada	Al realizar los pasos indicados se mostrará la pantalla de login, que mostrará únicamente un botón para poder loguearse en la aplicación a través de twitter.

Tabla 77. PR-07

Identificador	PR-08
Aplicación	Alumno
Descripción	Al pulsar el botón para loguearse con un nuevo usuario o loguearse por primera vez, la aplicación nos redireccionará a la página de autenticación de twitter, donde podremos introducir el usuario y la contraseña de la cuenta que queremos utilizar.
Precondiciones	-
Pasos	<ul style="list-style-type: none"> • Abrir la aplicación. • Pulsar el botón de login en twitter.
Salida esperada	Se mostrará la pantalla de autenticación de twitter. En ella podremos introducir el usuario y la contraseña con el que queremos realizar el login.

Tabla 78. PR-08

Identificador	PR-09
Aplicación	Alumno
Descripción	Al introducir un usuario y contraseña correctas y que pertenezcan a una cuenta de alumno, la aplicación mostrará la pantalla con las distintas asignaturas.
Precondiciones	-
Pasos	<ul style="list-style-type: none"> • Abrir la aplicación. • Pulsar el botón de login. • Introducir un usuario correspondiente a una cuenta de alumno.
Salida esperada	Se mostrará la actividad que contendrá las distintas asignaturas.

Tabla 79. PR-09

Identificador	PR-10
Aplicación	Alumno
Descripción	Durante el login, al introducir un usuario que es un profesor deberá mostrarse un aviso que indique que esa cuenta es una cuenta de profesor y que el login debe realizarse con una cuenta de alumno.
Precondiciones	En el servidor FTP se incluirá una asignatura que se corresponderá con la cuenta utilizada durante el login.
Pasos	<ul style="list-style-type: none"> • Abrir la aplicación. • Pulsar el botón de login. • Introducir un usuario que sea un profesor y su correspondiente contraseña.
Salida esperada	Pantalla de login en la que aparecerá un aviso que indicará “Esta cuenta es una cuenta de profesor. Para loguearse mediante esta cuenta descargue la aplicación de profesor”.

Tabla 80. PR-10

Identificador	PR-11
Aplicación	Alumno
Descripción	Al abrir la aplicación, cuando ya nos hemos logueado previamente con un usuario y accedemos a la pantalla de login, además del botón de login en twitter aparecerá un desplegable con todas las cuentas introducidas anteriormente y un botón de loguearse con un nuevo usuario.
Precondiciones	Haberse logueado con al menos un usuario y después haber cerrado sesión.
Pasos	<ul style="list-style-type: none"> • Abrir la aplicación.
Salida esperada	Pantalla de login que contendrá un listado de las cuentas con las que nos hemos logueado anteriormente, un botón para loguearse con uno de ellos y otro botón que permitirá loguearse con un nuevo usuario.

Tabla 81. PR-11

Identificador	PR-12
Aplicación	Alumno
Descripción	Al acceder a la aplicación cuando ya estamos logueados con una cuenta de alumno se mostrará directamente la pantalla con las distintas asignaturas en lugar de la pantalla de login.
Precondiciones	Estar logueado en la aplicación con una cuenta de alumno.
Pasos	<ul style="list-style-type: none"> • Abrir la aplicación.
Salida esperada	Se mostrará la interfaz que contendrá un listado con todas las asignaturas que están disponibles para el alumno con el que se ha realizado el login.

Tabla 82. PR-12

5.2 Pruebas de sistema: CU-02

A continuación se mostrarán las pruebas relacionadas con el caso de uso CU-02, que tendrán que ver con la visualización de las asignaturas de un alumno.

Identificador	PR-13
Aplicación	Alumno
Descripción	Al entrar en la aplicación estando logueado con un usuario se deberán visualizar únicamente las asignaturas que tiene dicho alumno.
Precondiciones	En el servidor FTP existirán tres asignaturas. El alumno pertenecerá únicamente a dos de ellas.
Pasos	<ul style="list-style-type: none">• Abrir la aplicación.• Loguearse con un usuario.
Salida esperada	En el listado de asignaturas que se muestra en pantalla deberá visualizarse las dos asignaturas a las que pertenece el alumno.

Tabla 83. PR-13

Identificador	PR-14
Aplicación	Alumno
Descripción	Si al entrar en la actividad de visualización de asignaturas el alumno no tiene ninguna asignatura disponible deberá mostrarse un mensaje en pantalla que lo indique.
Precondiciones	En el servidor FTP no existirá ninguna asignatura a la que pertenezca el alumno.
Pasos	<ul style="list-style-type: none">• Abrir la aplicación.• Loguearse con un usuario.
Salida esperada	Pantalla que mostrará en la parte derecha un mensaje que indique que el alumno no tiene ninguna asignatura disponible.

Tabla 84. PR-14

5.3 Pruebas de sistema: CU-03

A continuación se mostrarán las pruebas relacionadas con el caso de uso CU-03, que tendrán que ver con la visualización de las evaluaciones existentes en una asignatura.

Identificador	PR-15
Aplicación	Profesor
Descripción	Al entrar en la aplicación estando logueado con un usuario se visualizarán todas las evaluaciones existentes en la asignatura.
Precondiciones	En el servidor FTP, para la carpeta de la asignatura del profesor logueado deberán existir 3 evaluaciones.
Pasos	<ul style="list-style-type: none">• Abrir la aplicación.• Loguearse con un usuario.
Salida esperada	Pantalla que mostrará en la parte izquierda un listado con las tres evaluaciones existentes en la asignatura impartida.

Tabla 85. PR-15

Identificador	PR-16
Aplicación	Profesor
Descripción	Al entrar en la aplicación estando logueado se deberá diferencia entre las evaluaciones abiertas y las evaluaciones cerradas.
Precondiciones	En el servidor FTP, para la carpeta de la asignatura del profesor logueado, deberán existir tres evaluaciones. Dos de ellas se encontrarán abiertas y la otra se encontrará cerrada.
Pasos	<ul style="list-style-type: none">• Abrir la aplicación.• Loguearse con un usuario.
Salida esperada	En el listado de evaluaciones existentes aparecerá la palabra “Cerrada” bajo la evaluación que está cerrada y la palabra “Abierta” bajo las dos evaluaciones que están abiertas.

Tabla 86. PR-16

Identificador	PR-17
Aplicación	Profesor
Descripción	Al loguearse en la aplicación, si la asignatura del profesor que se ha logueado en la aplicación no tiene ninguna evaluación disponible se mostrará un mensaje en pantalla que indicará que no hay evaluaciones.
Precondiciones	En el servidor FTP existirá una asignatura que se corresponda con la cuenta con la que se ha logueado el profesor. En dicha asignatura no existirá ninguna evaluación.
Pasos	<ul style="list-style-type: none"> • Abrir la aplicación. • Loguearse con un usuario.
Salida esperada	Pantalla que mostrará en la parte derecha un mensaje que indicará que no hay evaluaciones disponibles.

Tabla 87. PR-17

Identificador	PR-18
Aplicación	Alumno
Descripción	Al seleccionar una asignatura se visualizarán todas las evaluaciones existentes para dicha asignatura.
Precondiciones	En el servidor FTP, para la carpeta de la asignatura seleccionada por el alumno deberán existir 3 evaluaciones.
Pasos	<ul style="list-style-type: none"> • Abrir la aplicación. • Loguearse con un usuario. • Seleccionar una asignatura.
Salida esperada	Pantalla que mostrará en la parte izquierda un listado con las tres evaluaciones existentes en la asignatura seleccionada.

Tabla 88. PR-18

Identificador	PR-19
Aplicación	Alumno
Descripción	Al pulsar sobre una asignatura y mostrarse las distintas evaluaciones se deberá diferenciar entre las evaluaciones abiertas, las cerradas, las realizadas y las no realizadas.
Precondiciones	En el servidor FTP, para la carpeta de la asignatura seleccionada por el alumno deberán existir cuatro evaluaciones. Una de ellas se encontrará cerrada y no realizada por el alumno, otra se encontrará cerrada y realizada por el alumno, otra se encontrará abierta y no realizada y la última se encontrará abierta y realizada.
Pasos	<ul style="list-style-type: none"> • Abrir la aplicación. • Loguearse con un usuario. • Seleccionar una asignatura.
Salida esperada	En el listado de evaluaciones existentes aparecerá un listado de con las cuatro evaluaciones de la asignatura. La primera y la segunda aparecerán como cerradas, la tercera como no realizada y la cuarta como realizada.

Tabla 89. PR-19

Identificador	PR-20
Aplicación	Alumno
Descripción	Cuando un alumno seleccione una asignatura que no tiene evaluaciones deberá mostrarse un mensaje indicándolo.
Precondiciones	En la asignatura correspondiente a la cuenta con la que se ha realizado el login no existirá ninguna evaluación.
Pasos	<ul style="list-style-type: none"> • Abrir la aplicación. • Loguearse con un usuario. • Seleccionar una asignatura.
Salida esperada	Pantalla que mostrará en la parte derecha un mensaje que indicará que no hay evaluaciones disponibles.

Tabla 90. PR-20

Identificador	PR-21
Aplicación	Alumno
Descripción	Cuando un alumno selecciona una evaluación que no tiene ninguna evaluación disponible se mostrará un mensaje en pantalla que indicará que no hay evaluaciones.
Precondiciones	En el servidor FTP no existirá ninguna evaluación para la asignatura seleccionada por el alumno.
Pasos	<ul style="list-style-type: none"> • Abrir la aplicación. • Seleccionar una asignatura.
Salida esperada	Pantalla que mostrará en la parte derecha un mensaje que indicará que no hay evaluaciones disponibles.

Tabla 91. PR.21

5.4 Pruebas de sistema: CU-04

A continuación se mostrarán las pruebas relacionadas con el caso de uso CU-04, que tendrán que ver con la visualización de las grupos existentes para una evaluación dada.

Identificador	PR-22
Aplicación	Profesor
Descripción	Al pulsar sobre una evaluación del listado de evaluaciones, se mostrará un listado de todos los grupos de dicha evaluación.
Precondiciones	En el servidor FTP se encontrará un fichero con los distintos grupos para la evaluación seleccionada.
Pasos	<ul style="list-style-type: none"> • Abrir la aplicación. • Seleccionar la evaluación deseada.
Salida esperada	Pantalla que mostrará en la parte izquierda un listado con todos los grupos de la evaluación seleccionada.

Tabla 92. PR-22

Identificador	PR-23
Aplicación	Profesor
Descripción	Tras seleccionar una evaluación deberá diferenciarse entre los grupos que han sido evaluados y los que no.
Precondiciones	En el servidor FTP se encontrará un fichero con los distintos grupos para la evaluación seleccionada. Se habrá realizado la evaluación de uno de los grupos.
Pasos	<ul style="list-style-type: none"> • Abrir la aplicación. • Seleccionar la evaluación deseada.
Salida esperada	Pantalla que mostrará en la parte izquierda un listado con todos los grupos de la evaluación seleccionada. Para el grupo del que se ha realizado la evaluación se visualizará "Evaluado", para el resto de grupos se visualizará "No evaluado".

Tabla 93. PR-23

5.5 Pruebas de sistema: CU-05

A continuación se mostrarán las pruebas relacionadas con el caso de uso CU-05, que tendrán que ver con la realización de las evaluaciones.

Identificador	PR-24
Aplicación	Profesor
Descripción	Al pulsar sobre un grupo no evaluado aparecerá la pantalla de evaluaciones.
Precondiciones	Existe al menos un grupo que no ha sido evaluado.
Pasos	<ul style="list-style-type: none"> • Abrir la aplicación. • Seleccionar la evaluación deseada. • Seleccionar el grupo deseado.
Salida esperada	Aparecerá la pantalla de evaluaciones. En esta pantalla se verá en la parte izquierda un listado con los grupos de criterios, y en la derecha podrán seleccionarse los criterios cumplidos del primer grupo de criterios.

Tabla 94. PR-24

Identificador	PR-25
Aplicación	Profesor
Descripción	Al pulsar sobre un nuevo grupo de criterios del listado aparecerán los criterios correspondientes a dicho grupo de criterios.
Precondiciones	Existe al menos un grupo de alumnos que no ha sido evaluado.
Pasos	<ul style="list-style-type: none"> • Abrir la aplicación. • Seleccionar una evaluación deseada. • Seleccionar el grupo deseado. • Pulsar sobre un grupo de criterios del listado.
Salida esperada	En la parte derecha de la pantalla de evaluación se verán los distintos criterios del grupo de criterios seleccionado y podrán ser marcados o desmarcados en función de si el criterio ha sido o no cumplido.

Tabla 95. PR-25

Identificador	PR-26
Aplicación	Profesor
Descripción	Al pulsar sobre el botón enviar, la evaluación realizada por el profesor será enviada al servidor FTP.
Precondiciones	Existe al menos un grupo que no ha sido evaluado.
Pasos	<ul style="list-style-type: none"> • Abrir la aplicación. • Seleccionar una evaluación deseada. • Seleccionar el grupo deseado. • Rellenar la evaluación. • Pulsar el botón enviar.
Salida esperada	En pantalla deberá aparecer un mensaje que indique que la evaluación ha sido enviada. Además en el servidor FTP, en la carpeta correspondiente al grupo evaluado deberá haberse creado un fichero que contenga los datos de la evaluación enviada.

Tabla 96. PR-26

Identificador	PR-27
Aplicación	Alumno
Descripción	Al pulsar sobre una evaluación no realizada se mostrará la pantalla desde la que los alumnos pueden evaluar a sus compañeros.
Precondiciones	Existe al menos una evaluación abierta y que no ha sido realizada por el alumno.
Pasos	<ul style="list-style-type: none"> • Abrir la aplicación. • Seleccionar una asignatura deseada. • Seleccionar la evaluación que se desea realizar.
Salida esperada	Aparecerá la pantalla de evaluaciones. En esta pantalla se verá en la parte izquierda un listado con los grupos de criterios, y en la parte derecha el primero de los criterios del primer grupo de criterios. Este criterio podrá marcarse y desmarcarse.

Tabla 97. PR-27

Identificador	PR-28
Aplicación	Alumno
Descripción	Al marcar un criterio como cumplido, aparecerá en la parte inferior de la mitad derecha de la pantalla una lista que incluirá a los alumnos del grupo al que pertenece el alumno. Los alumnos podrán ser marcados y desmarcados en función de si han participado o no en el cumplimiento del criterio.
Precondiciones	Existe al menos una evaluación que no ha sido realizada.
Pasos	<ul style="list-style-type: none"> • Abrir la aplicación. • Seleccionar la asignatura deseada. • Seleccionar la evaluación deseada. • Marcar el criterio como cumplido.
Salida esperada	En la parte inferior de la pantalla se mostrarán los alumnos que forman parte del grupo al que pertenece el alumno y podrán ser marcados y desmarcados.

Tabla 98. PR-28

Identificador	PR-29
Aplicación	Alumno
Descripción	Al pulsar sobre el botón siguiente de un grupo de criterios, se mostrará el siguiente criterio de dicho grupo.
Precondiciones	El grupo de criterios seleccionado contiene más de un criterio.
Pasos	<ul style="list-style-type: none"> • Abrir la aplicación. • Seleccionar una asignatura. • Seleccionar la evaluación que se desea realizar. • Pulsar el botón siguiente de la pantalla de evaluación.
Salida esperada	Se mostrará por pantalla la descripción del segundo criterio del grupo de criterios seleccionado y éste podrá ser marcado o desmarcado.

Tabla 99. PR-29

Identificador	PR-30
Aplicación	Alumno
Descripción	Al pulsar sobre el botón anterior de un grupo de criterios, se mostrará el criterio anterior de dicho grupo de criterios.
Precondiciones	El grupo de criterios seleccionado contiene más de un criterio.
Pasos	<ul style="list-style-type: none"> • Abrir la aplicación. • Seleccionar la asignatura deseada. • Seleccionar la evaluación que se desea realizar. • Pulsar el botón siguiente. • Pulsar el botón anterior.
Salida esperada	Se mostrará por pantalla la descripción del primer criterio del grupo de criterios seleccionado y éste podrá ser marcado o desmarcado.

Tabla 100. PR-30

Identificador	PR-31
Aplicación	Alumno
Descripción	Al hacer desplazamiento lateral hacia la izquierda en la pantalla de criterios, se mostrará el siguiente criterio.
Precondiciones	El grupo de criterios seleccionado contiene más de un criterio.
Pasos	<ul style="list-style-type: none"> • Abrir la aplicación. • Seleccionar la asignatura deseada. • Seleccionar la evaluación que se desea realizar. • Hacer desplazamiento lateral hacia la izquierda.
Salida esperada	Se mostrará por pantalla la descripción del segundo criterio del grupo de criterios seleccionado y éste podrá ser marcado o desmarcado.

Tabla 101. PR-31

Identificador	PR-32
Aplicación	Alumno
Descripción	Al hacer desplazamiento lateral hacia la derecha en la pantalla de criterios, se mostrará el criterio anterior.
Precondiciones	El grupo de criterios seleccionado contiene más de un criterio.
Pasos	<ul style="list-style-type: none"> • Abrir la aplicación. • Seleccionar la asignatura deseada. • Seleccionar la evaluación que se desea realizar. • Hacer desplazamiento lateral hacia la izquierda. • Hacer desplazamiento lateral hacia la derecha.
Salida esperada	Se mostrará por pantalla la descripción del primer criterio del grupo de criterios seleccionado y éste podrá ser marcado o desmarcado.

Tabla 102. PR-32

Identificador	PR-33
Aplicación	Alumno
Descripción	Al pulsar sobre el botón enviar, la evaluación realizada por el alumno será enviada al servidor FTP.
Precondiciones	Existe al menos una evaluación que no ha sido realizada.
Pasos	<ul style="list-style-type: none"> • Abrir la aplicación. • Seleccionar una asignatura. • Seleccionar la evaluación deseada. • Rellenar la evaluación. • Pulsar el botón enviar.
Salida esperada	En pantalla deberá aparecer un mensaje que indique que la evaluación ha sido enviada. Además en el servidor FTP, en la carpeta correspondiente al grupo evaluado, deberá haberse creado un fichero que contenga los datos de la evaluación enviada.

Tabla 103. PR-33

5.6 Pruebas de sistema: CU-06

A continuación se mostrarán las pruebas relacionadas con el caso de uso CU-06, que tendrán que ver con la visualización de las evaluaciones realizadas.

Identificador	PR-34
Aplicación	Profesor
Descripción	Al pulsar sobre un grupo evaluado, el profesor deberá visualizar los datos que introdujo en el momento en el que realizó la evaluación.
Precondiciones	El profesor debe haber realizado al menos una evaluación.
Pasos	<ul style="list-style-type: none"> • Abrir la aplicación. • Seleccionar una evaluación. • Seleccionar un grupo evaluado.
Salida esperada	El profesor deberá ver en pantalla los mismos datos que introdujo cuando realizó la evaluación del grupo.

Tabla 104. PR-22

Identificador	PR-35
Aplicación	Profesor
Descripción	No podrán modificarse los datos de una evaluación ya realizada.
Precondiciones	El profesor debe haber realizado al menos una evaluación.
Pasos	<ul style="list-style-type: none"> • Abrir la aplicación. • Seleccionar una evaluación. • Seleccionar un grupo evaluado. • Tratar de marcar un criterio que no esté marcado.
Salida esperada	El criterio que se ha tratado de marcar permanecerá desmarcado.

Tabla 105. PR-35

Identificador	PR-36
Aplicación	Profesor
Descripción	Una evaluación ya realizada no podrá ser enviada de nuevo.
Precondiciones	El profesor debe haber realizado al menos una evaluación.
Pasos	<ul style="list-style-type: none"> • Abrir la aplicación. • Seleccionar una evaluación deseada. • Seleccionar un grupo evaluado.
Salida esperada	El botón de enviar evaluación no deberá mostrarse en pantalla.

Tabla 106. PR-36

Identificador	PR-37
Aplicación	Alumno
Descripción	Al pulsar sobre una evaluación realizada el alumno visualizará los datos que introdujo en el momento en el que la realizó.
Precondiciones	El alumno debe haber realizado al menos una evaluación.
Pasos	<ul style="list-style-type: none"> • Abrir la aplicación. • Seleccionar una asignatura. • Seleccionar una evaluación.
Salida esperada	El alumno deberá ver en pantalla los mismos datos que rellenó cuando realizó la evaluación del grupo.

Tabla 107. PR-37

Identificador	PR-38
Aplicación	Alumno
Descripción	No podrán modificarse los datos de una evaluación que ya ha sido realizada.
Precondiciones	El alumno debe haber realizado al menos una evaluación de los miembros de su grupo.
Pasos	<ul style="list-style-type: none"> • Abrir la aplicación. • Seleccionar una asignatura. • Seleccionar una evaluación ya realizada. • Tratar de marcar un criterio que no esté marcado. • Tratar de desmarcar un alumno que esté marcado.
Salida esperada	El criterio que se ha tratado de marcar permanecerá desmarcado y el alumno que se ha tratado de desmarcar permanecerá marcado.

Tabla 108.PR-35

Identificador	PR-39
Aplicación	Alumno
Descripción	Una evaluación ya realizada por el alumno no podrá ser enviada de nuevo.
Precondiciones	El alumno debe haber realizado al menos una evaluación de los miembros de su grupo.
Pasos	<ul style="list-style-type: none"> • Abrir la aplicación. • Seleccionar una asignatura. • Seleccionar una evaluación ya realizada.
Salida esperada	El botón de enviar evaluación no deberá mostrarse en la pantalla de la evaluación.

Tabla 109. PR-39

5.7 Pruebas de sistema: CU-07

A continuación se mostrarán las pruebas relacionadas con el caso de uso CU-07, que estarán relacionadas con cerrar una evaluación.

Identificador	PR-40
Aplicación	Profesor
Descripción	Al mantener pulsada una evaluación abierta aparecerá la opción de cerrar evaluación.
Precondiciones	El profesor deberá tener al menos una evaluación abierta.
Pasos	<ul style="list-style-type: none">• Abrir la aplicación.• Realizar una pulsación larga sobre una evaluación no cerrada.
Salida esperada	Aparecerá una opción que permitirá cerrar una evaluación sobre la que se ha realizado la pulsación larga.

Tabla 110. PR-40

Identificador	PR-41
Aplicación	Profesor
Descripción	Al pulsar sobre la opción cerrar evaluación, si todos los grupos han sido evaluados, la evaluación se cerrará.
Precondiciones	El profesor debe haber evaluado a todos los grupos de la evaluación que pretende cerrar.
Pasos	<ul style="list-style-type: none">• Abrir la aplicación.• Realizar una pulsación larga sobre una evaluación no cerrada.• Pulsar sobre la opción cerrar evaluación.
Salida esperada	La evaluación que antes aparecía como abierta ahora aparecerá como cerrada.

Tabla 111. PR-41

Identificador	PR-42
Aplicación	Profesor
Descripción	El profesor no podrá modificar una evaluación cerrada.
Precondiciones	El profesor deberá tener al menos una evaluación cerrada.
Pasos	<ul style="list-style-type: none"> • Abrir la aplicación. • Seleccionar una evaluación cerrada. • Seleccionar un grupo. • Tratar de marcar un criterio que no esté marcado.
Salida esperada	El criterio que se ha tratado de marcar permanecerá desmarcado.

Tabla 112. PR-42

Identificador	PR-43
Aplicación	Alumno
Descripción	Al abrir una evaluación cerrada no podrá ser modificada.
Precondiciones	El alumno deberá tener al menos una evaluación cerrada.
Pasos	<ul style="list-style-type: none"> • Abrir la aplicación. • Seleccionar una asignatura. • Seleccionar una evaluación cerrada. • Tratar de marcar un criterio que no esté marcado.
Salida esperada	El criterio que se ha tratado de marcar permanecerá desmarcado.

Tabla 113. PR-43

Identificador	PR-44
Aplicación	Alumno
Descripción	No podrá enviarse una evaluación que se encuentra cerrada.
Precondiciones	El alumno deberá tener al menos una evaluación cerrada.
Pasos	<ul style="list-style-type: none"> • Abrir la aplicación. • Seleccionar una asignatura. • Seleccionar una evaluación cerrada.
Salida esperada	No se visualizará el botón de enviar evaluación.

Tabla 114. PR-44

5.8 Pruebas de sistema: CU-08

A continuación se mostrarán las pruebas relacionadas con el caso de uso CU-08, que estarán relacionadas con la visualización de la nota de los distintos grupos en una evaluación cuya nota esté disponible.

Identificador	PR-45
Aplicación	Profesor
Descripción	Al pulsar sobre la pestaña “Notas” en la pantalla de visualización de los grupos de una evaluación se visualizará un listado con todos los grupos de alumnos que hayan sido evaluados por el profesor.
Precondiciones	El profesor deberá haber evaluado al menos a un grupo.
Pasos	<ul style="list-style-type: none">• Abrir la aplicación.• Seleccionar una evaluación.• Pulsar la pestaña Notas.
Salida esperada	Aparecerá un listado con todos los grupos que hayan sido evaluados por el profesor.

Tabla 115. PR-45

Identificador	PR-46
Aplicación	Profesor
Descripción	Si al pulsar sobre la pestaña “Notas” no hay ninguna nota disponible se visualizará un mensaje que indique que no hay notas disponibles.
Precondiciones	El profesor no deberá haber evaluado a ningún grupo de la evaluación.
Pasos	<ul style="list-style-type: none">• Abrir la aplicación.• Seleccionar una evaluación.• Pulsar la pestaña Notas.
Salida esperada	Aparecerá un mensaje indicando que no hay notas disponibles.

Tabla 116. PR-46

Identificador	PR-47
Aplicación	Profesor
Descripción	Al pulsar sobre un grupo del listado de grupos evaluados se mostrara una pantalla que permitirá ver la nota obtenida por los alumnos del grupo seleccionado.
Precondiciones	El profesor deberá haber evaluado al menos a un grupo.
Pasos	<ul style="list-style-type: none"> • Abrir la aplicación. • Seleccionar una evaluación. • Pulsar la pestaña “Notas”. • Seleccionar un grupo.
Salida esperada	En la parte izquierda de la pantalla se podrá seleccionarse el alumno del grupo del que se desea visualizar la nota y en la parte derecha se visualizará la nota obtenida dicho alumno.

Tabla 117. PR-47

5.9 Pruebas de sistema: CU-09

A continuación se mostrarán las pruebas relacionadas con el caso de uso CU-09, que estarán relacionadas con la visualización del listado de alumnos de la asignatura impartida por el profesor.

Identificador	PR-48
Aplicación	Profesor
Descripción	Al pulsar la pestaña notas de la pantalla que muestra el listado de evaluaciones aparecerá el listado de alumnos de la asignatura.
Precondiciones	-
Pasos	<ul style="list-style-type: none"> • Abrir la aplicación. • Pulsar la pestaña “Notas”
Salida esperada	Aparecerá un listado con todos los alumnos de la asignatura

Tabla 118. PR-48

5.10 Pruebas de sistema: CU-10

A continuación se mostrarán las pruebas relacionadas con el caso de uso CU-10, que estarán relacionadas con la visualización de las notas obtenidas por un alumno.

Identificador	PR-49
Aplicación	Profesor
Descripción	Al pulsar sobre un alumno de la pestaña notas se mostrará un listado de las evaluaciones cuya nota está disponible para el alumno seleccionado.
Precondiciones	Existe al menos una nota disponible para el alumno seleccionado.
Pasos	<ul style="list-style-type: none">• Abrir la aplicación.• Pulsar la pestaña “Notas”• Seleccionar un alumno
Salida esperada	Aparecerá un listado con las evaluaciones cuya nota está disponible para el alumno.

Tabla 119. PR-49

Identificador	PR-50
Aplicación	Profesor
Descripción	Al pulsar sobre un alumno de la pestaña notas, si el alumno no tiene ninguna nota disponible, se mostrará un mensaje que indique que no hay notas disponibles.
Precondiciones	El alumno seleccionado no tiene notas disponibles.
Pasos	<ul style="list-style-type: none">• Abrir la aplicación.• Pulsar la pestaña “Notas”.• Seleccionar un alumno.
Salida esperada	Aparecerá un mensaje indicando que no hay notas disponibles.

Tabla 120. PR-50

Identificador	PR-51
Aplicación	Profesor
Descripción	Al pulsar sobre una evaluación del listado que muestra las evaluaciones que tienen una nota disponible para un alumno dado, se mostrará la nota obtenida por dicho alumno en la evaluación seleccionada.
Precondiciones	El alumno seleccionado tiene al menos una nota disponible para la asignatura impartida por el profesor.
Pasos	<ul style="list-style-type: none"> • Abrir la aplicación. • Pulsar la pestaña “Notas”. • Seleccionar un alumno. • Seleccionar una evaluación.
Salida esperada	En la parte derecha de la pantalla aparecerá la nota obtenida por el alumno en la evaluación seleccionada.

Tabla 121. PR-51

Identificador	PR-52
Aplicación	Alumno
Descripción	Al pulsar sobre la pestaña notas en la pantalla que muestra el listado de evaluaciones, el alumno podrá ver un listado con todas las evaluaciones cuya nota está disponible.
Precondiciones	El alumno tiene al menos una nota disponible.
Pasos	<ul style="list-style-type: none"> • Abrir la aplicación. • Seleccionar una asignatura. • Pulsar sobre la pestaña “Notas”.
Salida esperada	Se mostrará un listado con todas las evaluaciones cuya nota está disponible para el alumno.

Tabla 122. PR-52

Identificador	PR-53
Aplicación	Profesor
Descripción	Al pulsar sobre una evaluación de la pestaña notas, si el alumno no tiene ninguna nota disponible, se mostrará un mensaje que indique que no hay notas disponibles.
Precondiciones	El alumno no tiene ninguna nota disponible.
Pasos	<ul style="list-style-type: none"> • Abrir la aplicación. • Seleccionar una asignatura. • Pulsar la pestaña “Notas”.
Salida esperada	Aparecerá un mensaje en la pantalla indicando que no hay notas disponibles.

Tabla 123. PR-53

Identificador	PR-54
Aplicación	Alumno
Descripción	Al pulsar sobre una evaluación del listado que muestra las evaluaciones que tienen una nota disponible, se mostrará la nota obtenida por el alumno en la evaluación seleccionada.
Precondiciones	El alumno seleccionado tiene al menos una nota disponible para la asignatura impartida por el profesor.
Pasos	<ul style="list-style-type: none"> • Abrir la aplicación. • Pulsar la pestaña “Notas”. • Seleccionar un alumno. • Seleccionar una evaluación.
Salida esperada	En la parte derecha de la pantalla aparecerá la nota obtenida por el alumno en la evaluación seleccionada. En la parte izquierda se mostrará un listado en el que se podrá escoger entre visualizar la nota del alumno o la nota del grupo al que pertenece.

Tabla 124. PR-54

5.11 Pruebas de sistema: CU-11

A continuación se mostrarán las pruebas relacionadas con el caso de uso CU-11, y estarán relacionadas con cerrar sesión en la aplicación.

Identificador	PR-55
Aplicación	Profesor
Descripción	La aplicación permite cerrar sesión.
Precondiciones	El profesor deberá estar logueado en la aplicación.
Pasos	<ul style="list-style-type: none">• Abrir la aplicación.• Pulsar el botón de opciones del terminal.• Pulsar sobre la opción “Cerrar Sesión”.
Salida esperada	Se mostrará la pantalla de login que permitirá al usuario loguearse de nuevo.

Tabla 125. PR-55

Identificador	PR-56
Aplicación	Alumno
Descripción	La aplicación permite cerrar sesión.
Precondiciones	El alumno deberá estar logueado en la aplicación.
Pasos	<ul style="list-style-type: none">• Abrir la aplicación.• Pulsar el botón de opciones del terminal.• Pulsar sobre la opción “Cerrar Sesión”.
Salida esperada	Se mostrará la pantalla de login que permitirá al usuario loguearse de nuevo.

Tabla 126. PR-56

Capítulo VI.

Planificación

En este capítulo se realizará la planificación llevada a cabo durante el proyecto. Como ya se ha explicado en el segundo capítulo, la planificación será llevada a cabo mediante la metodología Scrum. Mediante este método de planificación el trabajo es estructurado en ciclos de trabajo llamados sprints.

Puesto que la planificación realizada será una planificación adaptativa, no se realiza una planificación inicial que incluya todas las etapas del proyecto, sino que las tareas a llevar a cabo en cada sprint son determinadas una vez se ha finalizado el sprint anterior.

Por tanto, a lo largo de este capítulo se describirán las distintas iteraciones que se han llevado a cabo, los objetivos de cada iteración y las tareas llevadas a cabo para cumplir dichos objetivos.

El tiempo estimado para completar la aplicación es de 300 horas y la realización del proyecto se ha dividido en diez iteraciones. Las fechas de inicio y fin de cada iteración pueden observarse en la siguiente tabla:

Número de iteración	Fecha de inicio	Fecha de fin
1	02/09/2013	15/09/2013
2	16/09/2013	29/09/2013
3	30/09/2013	13/10/2013
4	14/10/2013	27/10/2013
5	28/10/2013	10/11/2013
6	11/11/2013	24/11/2013
7	25/11/2013	08/12/2013
8	09/12/2013	22/12/2013
9	23/12/2013	05/01/2013
10	06/01/2013	24/01/2013

Tabla 127. Planificación. Iteraciones

A continuación se detallarán las tareas llevadas a cabo en cada una de las iteraciones.

Iteración 1

El primer sprint o iteración transcurre desde el día 2 de septiembre de 2013 hasta el día 15 de septiembre de 2013.

Durante este sprint los objetivos marcados son la familiarización con el entorno de desarrollo y con la programación de aplicaciones Android. Para ello se instalará el software necesario, se consultarán manuales y se realizarán ejemplos prácticos de programación de este tipo de aplicaciones.

Iteración 2

Esta segunda iteración comienza el día 16 de septiembre y finaliza el 29 de septiembre.

En primer lugar se realizará la extracción de requisitos del sistema a través de una serie de reuniones con el tutor en las que se van concretando las distintas historias de usuario y se van aclarando las distintas dudas que pueden surgir al respecto.

Además se pretende también realizar un estudio de las aplicaciones existentes en el mercado que presenten características similares o que guarden algún tipo de relación con la aplicación que queremos desarrollar.

Iteración 3

El tercer sprint comienza el día 30 de septiembre y en él se pretende realizar el diseño de la aplicación que se quiere desarrollar. Al finalizar esta iteración se deberán presentar las distintas interfaces diseñadas para valorar posibles cambios o mejoras en las mismas.

Iteración 4

Esta cuarta iteración comenzará el día 14 de octubre y finalizará el día 27 de ese mismo mes.

En ella se realizará la implementación de las historias de usuario HU-01-P y HU-01-A, ambas relacionadas con el login en las aplicaciones de profesores y alumno.

Además deberá ponerse en funcionamiento el servicio web, dotándole de las funcionalidades necesarias para el login en la aplicación.

También deberán realizarse las pruebas necesarias para asegurar el funcionamiento de los módulos implementados.

Las tareas llevadas a cabo en esta iteración son las siguientes:

- En la aplicación de profesores se han llevado a cabo las siguientes tareas:
 - Creación de la clase LoginActivity.java, que mostrará la interfaz correspondiente al login de los profesores.
 - Creación de la clase DataBaseHandler, que gestionará la base de datos del terminal encargada del almacenamiento de los distintos usuarios.
 - Creación de la clase HTTPManager, encargada de la comunicación con el servidor web. En ella se han implementado los siguientes métodos:
 - serverIsOn().
 - tipoCuenta(String usuario).
- En la aplicación de los alumnos se han llevado a cabo las siguientes tareas:
 - Creación de la clase LoginActivity.java, que mostrará la interfaz correspondiente al login de los profesores.
 - Creación de la clase DataBaseHandler.java, que gestionará la base de datos del terminal encargada del almacenamiento de los distintos usuarios.
 - Creación de la clase HTTPManager.java, encargada de la comunicación con el servidor web. En ella se han implementado los siguientes métodos:
 - serverIsOn().
 - tipoCuenta(String usuario).
- En cuanto al servicio web se han llevado a cabo las siguientes tareas:
 - Creación y puesta en marcha del servicio web.

- Implementación del método `serverIsOn()`.
- Implementación del método `conectarFTP()`.
- Implementación del método `tipoCuenta()`.
- Implementación del método `listarCarpetas()`.
- Implementación del método `listarFicheros()`.

Iteración 5

Esta iteración comenzará el día 28 de octubre y finalizará el día 10 de noviembre.

En ella se completará la implementación de las historias de usuario HU-02-A (relacionada con mostrar las asignaturas de un alumno), HU-02-P, HU-03-P, HU-03-A y HU-04-A (relacionadas con la visualización de las evaluaciones disponibles y su estado).

Además deberán realizarse las pruebas necesarias para garantizar el funcionamiento de las funcionalidades implementadas.

Las tareas llevadas a cabo en esta iteración son las siguientes:

- En la aplicación de profesores se han llevado a cabo las siguientes tareas:
 - Creación de la clase `ListadoEvaluaciones.java`, que mostrará un listado de todas las evaluaciones disponibles para la asignatura impartida y el estado en el que se encuentran dichas evaluaciones.
 - Creación de la clase `JSONManager()`, que se encargará de crear los ficheros JSON que se enviarán al servidor FTP y que interpretará los ficheros JSON descargados desde el servidor.
 - Ampliación de las funcionalidades de la clase `HTTPManager.java`. En ella se han implementado los siguientes métodos:
 - `listarEvaluaciones()`.
 - `evaluacionCerrada()`.
- En la aplicación de los alumnos se han llevado a cabo las siguientes tareas:
 - Creación de la clase `ListadoAsignaturas.java`, que mostrará la interfaz en la que podrán visualizarse las distintas asignaturas que tiene el alumno.

- Creación de la clase ListadoEvaluaciones.java, que mostrará la interfaz en la que podrán visualizarse las distintas evaluaciones disponibles para la asignatura seleccionada.
- Creación de la clase JSONManager(), que se encargará de crear los ficheros JSON que se enviarán al servidor FTP y que interpretará los ficheros JSON descargados desde el servidor.
- Ampliación de las funcionalidades de la clase HTTPManager.java. En ella se han implementado los siguientes métodos:
 - listarAsignaturas().
 - listarEvaluaciones().
 - evaluacionCerrada().
- En cuanto al servicio web se han llevado a cabo las siguientes tareas:
 - Implementación del método listarAsignaturas().
 - Implementación del método listarEvaluaciones().
 - Implementación del método evaluacionCerrada().
 - Implementación del método evaluacionRealizada().

Iteración 6

Esta iteración comenzará el día 11 de noviembre y finalizará el día 24 de ese mismo mes.

En ella se completará la implementación de las historias de usuario HU-04-P, HU-05-P, HU-06-P. Estas historias de usuario hacen referencia a cerrar evaluaciones, visualización de grupos de alumnos y visualización del estado de la evaluación para cada grupo.

Además deberán realizarse las pruebas necesarias para garantizar el funcionamiento de las funcionalidades implementadas.

Las tareas llevadas a cabo en esta iteración son las siguientes:

- En la aplicación de profesores se han llevado a cabo las siguientes tareas:

- Creación de la clase ListadoGrupos.java, que mostrará un listado de todos los grupos existentes para la evaluación seleccionada y el estado (evaluado o no evaluado) de cada grupo.
- Ampliación de las funcionalidades de la clase HTTPManager.java. En ella se han implementado los siguientes métodos:
 - cerrarEvaluacion().
 - listarGrupos ().
 - miembrosGrupo().
 - evaluacionRealizada().
- En la aplicación de los alumnos no se ha llevado a cabo ningún cambio.
- En cuanto al servicio web se han llevado a cabo las siguientes tareas:
 - Implementación del método listarGrupos().
 - Implementación del método cerrarEvaluacion().
 - Implementación del método miembrosGrupo().

Iteración 7

Esta séptima iteración comenzará el día 25 de noviembre y finalizará el día 8 de diciembre.

En ella se llevará a cabo la implementación de las historias de usuario HU-07P, HU-08-P, HU-05-A, HU-06-A, todas ellas relacionadas con la realización de las evaluaciones y con la posterior visualización de los datos introducidos.

Además deberán realizarse las pruebas necesarias para garantizar el funcionamiento de las funcionalidades implementadas.

Las tareas llevadas a cabo en esta iteración son las siguientes:

- En la aplicación de profesores se han llevado a cabo las siguientes tareas:
 - Creación de la clase Evaluacion.java, que mostrará la interfaz que permitirá al profesor realizar las evaluaciones de los alumnos y visualizar los datos de una evaluación ya realizada.

- Ampliación de las funcionalidades de la clase HTTPManager.java. En ella se han implementado los siguientes métodos:
 - descargarCriterios().
 - miembrosGrupos().
 - enviarEvaluacion().
 - descargarEvaluacion().
- En la aplicación de los alumnos se han llevado a cabo las siguientes tareas:
 - Creación de la clase Evaluacion.java, que mostrará la interfaz que permitirá al alumno evaluar a sus compañeros de grupo y visualizar los datos de una evaluación ya realizada.
 - Ampliación de las funcionalidades de la clase HTTPManager.java. En ella se han implementado los siguientes métodos:
 - descargarCriterios().
 - obtenerGrupo().
 - miembrosGrupos().
 - enviarEvaluacion().
 - descargarEvaluacion().
- En cuanto al servicio web se han llevado a cabo las siguientes tareas:
 - Implementación del método listarGrupos().
 - Implementación del método cerrarEvaluacion().
 - Implementación del método miembrosGrupo().

Iteración 8

Esta iteración comenzará el día 9 de diciembre y finalizará el día 22 de diciembre.

En ella se completará la implementación de las historias de usuario HU-09P, HU-10-P, HU-07-A (relacionadas la visualización de las notas), HU-11-P y HU-08-A, (encargadas de cerrar sesión en la aplicación).

Además deberán realizarse las pruebas necesarias para garantizar el correcto funcionamiento de las funcionalidades implementadas.

Las tareas llevadas a cabo en esta iteración son las siguientes:

- En la aplicación de profesores se han llevado a cabo las siguientes tareas:
 - Creación de la clase `NotasAlumno.java`, que mostrará la interfaz mediante la que se podrán consultar todas las notas que ha obtenido un alumno en cada una de las evaluaciones
 - Creación de la clase `NotasEvaluaciones.java`, que mostrará la interfaz mediante la que se podrá consultar la nota que han obtenido cada grupo en una evaluación determinada.
 - Ampliación de la clase `listadoEvaluaciones.java` para incluir una nueva pestaña que muestre todos los alumnos de la asignatura.
 - Modificación de todas las actividades (excepto `Login.java`) para permitir cerrar la sesión en cada una de ellas.
 - Ampliación de las funcionalidades de la clase `HTTPManager.java`. En ella se han implementado los siguientes métodos:
 - `listarAlumnos()`.
 - `descargarNotas()`.
 - `obtenerGrupo()`.
 - `miembrosGrupos()`.
- En la aplicación de los alumnos se han llevado a cabo las siguientes tareas:
 - Creación de la clase `Notas.java`, que mostrará la interfaz mediante la que se podrá consultar la nota que ha obtenido cada grupo en una evaluación determinada.
 - Ampliación de la clase `listadoEvaluaciones.java` para incluir una nueva pestaña que muestre todos los alumnos de la asignatura.
 - Modificación de todas las actividades (excepto `Login.java`) para permitir cerrar la sesión en cada una de ellas.
 - Ampliación de las funcionalidades de la clase `HTTPManager.java`. En ella se han implementado los siguientes métodos:
 - `descargarNotas()`.
- En cuanto al servicio web se han llevado a cabo las siguientes tareas:
 - Implementación del método `descargarNotas()`.
 - Implementación del método `listarAlumnos()`.

Iteración 9

Esta cuarta iteración comenzará el día 23 de diciembre de 2013 y finalizará el día 5 de enero de 2014. Ha de tenerse en cuenta las festividades existentes durante estas dos semanas, por lo que el tiempo invertido en esta iteración será inferior al del resto.

Durante esta iteración se realizarán las pruebas de todo el sistema una vez el desarrollo de la aplicación ha sido finalizado. Además se corregirán los posibles errores que puedan surgir.

Iteración 10

Esta última iteración será la más larga y tendrá una duración de 21 días. Comenzará el día 7 de enero y finalizará el día 24.

Durante esta iteración se completará y se revisará la memoria del proyecto.

Capítulo VII.

Presupuesto

A lo largo de este capítulo se realizará una descripción detallada de los costes que ha supuesto la realización de este proyecto. En él se incluirá tanto el coste de personal como el coste del software y del hardware utilizados en el proyecto.

7.1 Costes de personal

En este apartado se mostrarán las personas que han intervenido en el proyecto, el perfil de cada una de estas personas, las horas que han dedicado a su realización, el coste/hora de cada uno de ellos, y el coste total de cada uno de ellos.

Nombre	Perfil	Horas invertidas	Coste/Hora	Coste total
Ángel Ávila Martín	Ingeniero junior	300	12€/h	3600€
Alberto Heredia García	Ingeniero superior	20	25€/h	500€
Javier García Guzmán	Ingeniero superior	12	25€/h	300€
			Total	4400€

Tabla 128. Costes de personal

7.2 Costes software

El software utilizado durante la realización del proyecto es gratuito. Sin embargo, para la publicación de la aplicación en Google Play es necesario crear un perfil de desarrollador android.

Nombre	Coste
Licencia de desarrollador Android	18'53€
Total	18'53€

Tabla 129. Costes software.

Por tanto, el coste del software utilizado es de 18'53 €.

7.3 Costes hardware

A continuación se mostrará el coste del hardware utilizado. Para ello habrá que tener en cuenta su precio, el tiempo que ha sido utilizado, y el periodo de amortización.

La fórmula utilizada para el cálculo del coste hardware será el siguiente:

$$\text{Coste} = \frac{\text{Tiempo de utilización}}{\text{Periodo de amortización}} \times \text{Precio}$$

El hardware utilizado durante la realización del proyecto ha sido el que se muestra a continuación:

Nombre	Precio	Periodo de amortización	Tiempo utilizado	Coste
Ordenador portátil Packard Bell EasyNote TJ66	550€	60 meses	6 meses	55€
Galaxy Tablet 3	180€	24 meses	6 meses	45€
			Total	100€

Tabla 130. Costes hardware

7.4 Resumen de costes

Si sumamos todos los costes calculados anteriormente obtenemos un total de 4518'53€.

Coste de personal	4400 €
Coste software	18'53 €
Coste hardware	100 €
Total	4518'53 €

Tabla 131. Tabla de costes

Capítulo VIII.

Conclusión

En este capítulo se mostrarán las conclusiones alcanzadas a partir de la realización del proyecto. En ellas se incluirán los resultados obtenidos en el desarrollo, las futuras líneas de ampliación y mejora de la aplicación y una opinión personal sobre las experiencias y el aprendizaje adquiridos durante su realización.

8.1 Resultados obtenidos

El resultado obtenido tras la realización del proyecto ha sido muy satisfactorio. Se ha cumplido con los objetivos establecidos, completando el desarrollo tanto de las aplicaciones de profesor y alumnos como el desarrollo y puesta en funcionamiento del servicio web.

La aplicación resultante permite tanto a profesores como a alumnos realizar evaluaciones multifuente. Esto facilitará en gran medida la corrección de prácticas por parte de los profesores, ya que el cálculo de notas será realizado de forma automática y los alumnos podrán ver las notas en sus aplicaciones en cuanto las prácticas hayan sido evaluadas.

8.2 Líneas de ampliación

Una vez finalizado el proyecto existen una serie de ideas que podrían llevarse a cabo en una futura ampliación de la aplicación y que no ha sido posible llevar a cabo durante este proyecto debido a la limitación de tiempo existente. Estas líneas de ampliación quedan definidas en este apartado.

8.2.1 Login a través de otras redes sociales

Actualmente la aplicación únicamente permite realizar login a través de una cuenta de Twitter. Sin embargo, es probable que existan alumnos que no cuenten con una cuenta en esta red social, por lo que es recomendable que el login pueda realizarse a través de otros medios.

Debido a ello, se propone como una posible ampliación el permitir el login a través de otras redes sociales, como por ejemplo, Facebook.

8.2.2 Gestionar datos desde la aplicación

La aplicación permite actualmente que tanto profesores como alumnos realicen la evaluación de los distintos miembros que forman parte de un grupo.

Sin embargo, para que esto sea posible los profesores deberán haber subido anteriormente al servidor FTP toda la información necesaria, es decir, los alumnos que forman parte de una asignatura, los grupos que forman parte de una evaluación o los criterios que son evaluados al realizar la evaluación.

Se propone por tanto ampliar la funcionalidad de la aplicación para permitir a los profesores subir toda esta información desde la propia aplicación, sin necesidad de realizarlo a través de su ordenador u otra aplicación móvil. De este modo se permitiría al profesor crear los siguientes ficheros desde la aplicación:

- Fichero de alumnos de una asignatura.
- Ficheros de grupos de una evaluación.
- Fichero de criterios existentes en una evaluación.

8.2.3 Notificaciones push

Actualmente, un alumno se dará cuenta de que existe una evaluación disponible cuando esto sea comunicado por el profesor en clase o cuando el alumno acceda a la aplicación para comprobar si hay alguna novedad en la misma. Para poder consultar la nota sucede lo mismo, el alumno tiene que entrar en la aplicación y comprobar si la nota está ya disponible.

Con esta nueva ampliación se permitiría a los profesores avisar a los alumnos de que existe una nueva evaluación o de que hay una nueva nota disponible. De este modo, cuando los profesores envíen un aviso, creen una nueva evaluación o cierren una evaluación, los alumnos recibirán una notificación push en su dispositivo móvil o tablet. De este modo, los alumnos podrían saber de forma instantánea cuando se ha creado una nueva evaluación en la asignatura o cuando hay una nueva nota disponible.

8.3 Opinión personal

En este apartado se incluirán la opinión personal obtenida durante la creación de este proyecto, el aprendizaje obtenido a través de su realización y los distintos problemas que han surgido durante su desarrollo.

La creación de este proyecto ha sido un gran reto para mí. Es la primera vez que he creado una aplicación de estas características sin la ayuda de los compañeros que me han acompañado durante el transcurso de la carrera. Sin embargo, no ha sido tan complicado como esperaba en un principio y, a pesar de los problemas que han podido surgir durante el desarrollo, el proyecto ha seguido adelante y se ha finalizado dentro de los plazos esperados.

Una gran motivación para la realización de este proyecto ha sido la realización de la aplicación para dispositivos móviles android. Este tipo de dispositivos han llegado a convertirse en los dispositivos móviles más vendidos en todo el mundo y la programación de sus aplicaciones se encuentra en auge en este momento. El aprendizaje obtenido en la plataforma android me será probablemente de gran utilidad en un futuro.

Además de la programación en android, he aprendido también en poner en marcha un web service capaz de atender las peticiones realizadas al mismo a través de otra aplicación, lo que probablemente también me sea de gran utilidad.

En cuanto a los problemas encontrados, destaca el desconocimiento de la programación en android, ya que, aunque ya había programado anteriormente una pequeña aplicación en esta plataforma, el desconocimiento sobre las distintas opciones existentes era muy grande.

Como conclusión final he de decir que estoy ampliamente satisfecho con el trabajo realizado y que la programación de aplicaciones móviles me ha resultado útil y bastante entretenida, por lo que espero poder seguir programando este tipo de aplicaciones en un futuro cercano, creándolas no únicamente para el sistema operativo android sino también para IOS u otros sistemas operativos.

Capítulo IX.

Bibliografía

Referencias

Emulador

[1] <http://es.wikipedia.org/wiki/Emulador>

Evaluación multifuente

[2] <http://www.redalyc.org/articulo.oa?id=231317624003>

Metodología ágil

[3] http://es.wikipedia.org/wiki/Desarrollo_%C3%A1gil_de_software

Product Backlog

[4] <http://diegoforever.wikispaces.com/SEMINARIO+EN+LA+PRACTICA>

Servidor FTP

[5] <http://servidorftp.es>

Sprint

[6] <http://www.softeng.es/es-es/empresa/metodologias-de-trabajo/metodologia-scrum/proceso-roles-de-scrum.html>

PLE y PLN

[7] A. Heredia, J.A. Cuadrado Mingo, A. Amescua, J. Garcia-Guzman: "Exploring the Use of Social Identities for Sharing Knowledge in the Learning Process". Proceedings of the 5th annual International Conference on Education and New Learning Technologies (EDULEARN), Barcelona, España, 1-3 Julio 2013.

[8] Dabbagh, N. and Kitsantas, A. (2012) Personal Learning Environments, social media and selfregulated learning: A natural formula for connecting formal and informal learning. The Internet and Higher Education.

[9] Chatti, M.A., Agustiawan, M. R, Jarke, M. and Specht, M. (2010). Toward a Personal Learning Enviroment Framework. International Journal of Virtual and Personal Learning.

[10] <http://tallerple.wordpress.com/1-ple/>

[11] <http://www.slideshare.net/rcastillo72/tutorial-ple-pln-roy-castillo>

[12] Milligan, C.D., Beauvoir, P., Johnson, M. W., Sharples, P., Wilson, S. and Liber, O. (2006). Developing a Reference Model to Describe the Personal Learning Enviroment. In: Nejd, W. and Tochtermann, J. (eds.) EC-TEL 2006. LNCS, vol 4227., pp. 506- 511. Springer, Heridelberg.

[13] J.A. Cuadrado Mingo: "Plataforma para el uso de Twitter en la preparación e impartición de clases presenciales". Trabajo Fin de Master. Universidad Carlos III de Madrid. 2013.

Venta de dispositivos móviles por sistema operativo.

[14] <http://techland.time.com/2013/04/16/ios-vs-android/>

Android

[15] <http://es.wikipedia.org/wiki/Android>

Java

[16] [http://es.wikipedia.org/wiki/Java_\(lenguaje_de_programación\)](http://es.wikipedia.org/wiki/Java_(lenguaje_de_programación))

Eclipse

[17] <http://www.electronicweekly.com/eyes-on-android/what-is/what-iseclipse-2012-01/>

Scrum

[18] <http://es.wikipedia.org/wiki/Scrum>

[19] <http://www.softeng.es/es-es/empresa/metodologias-de-trabajo/metodologia-scrum.html>

Historias de usuario

[20] http://es.wikipedia.org/wiki/Historias_de_usuario

Casos de uso

[21] http://es.wikipedia.org/wiki/Caso_de_uso

Diagramas de secuencia

[22] http://es.wikipedia.org/wiki/Diagrama_de_secuencia